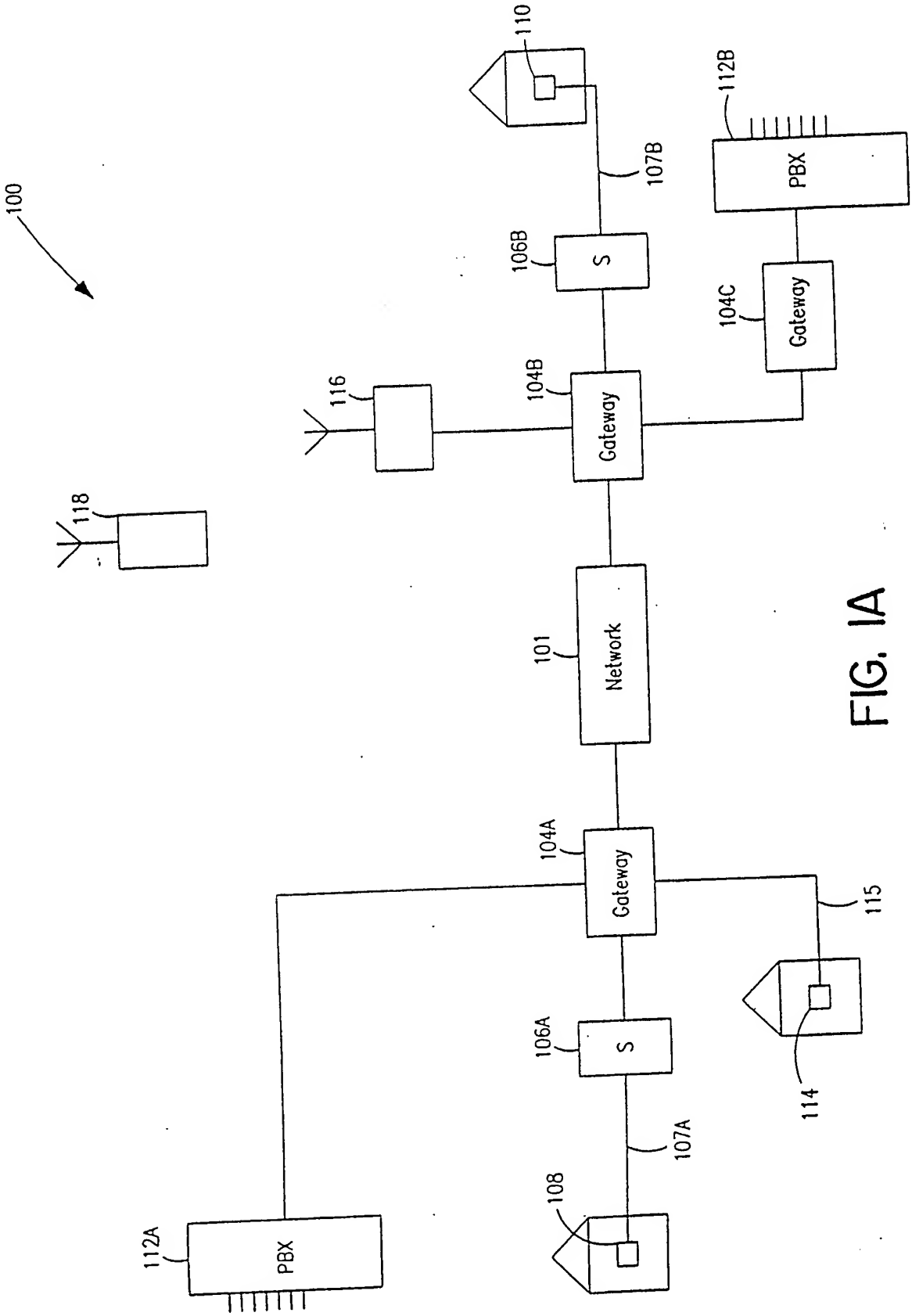


FIG. 1A

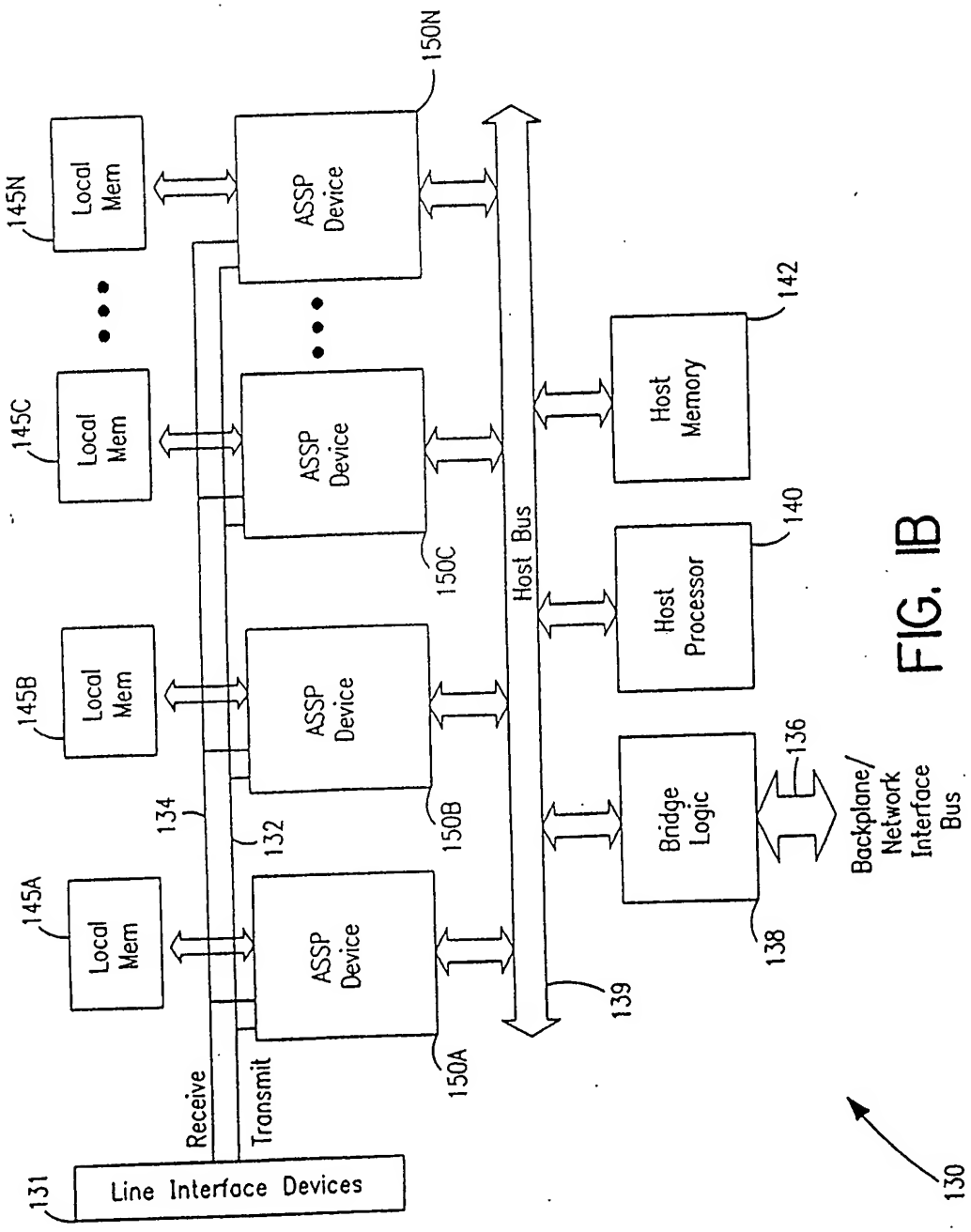

150N
145N
Local Mem
ASSP Device
142
Host Memory
145C
Local Mem
ASSP Device
150C
Host Bus
140
Host Processor
145B
Local Mem
ASSP Device
150B
134
132
136
Bridge Logic
Backplane/ Network Interface Bus
145A
Local Mem
ASSP Device
150A
139
138
Receive
Transmit
131
Line Interface Devices
130

FIG. 1B

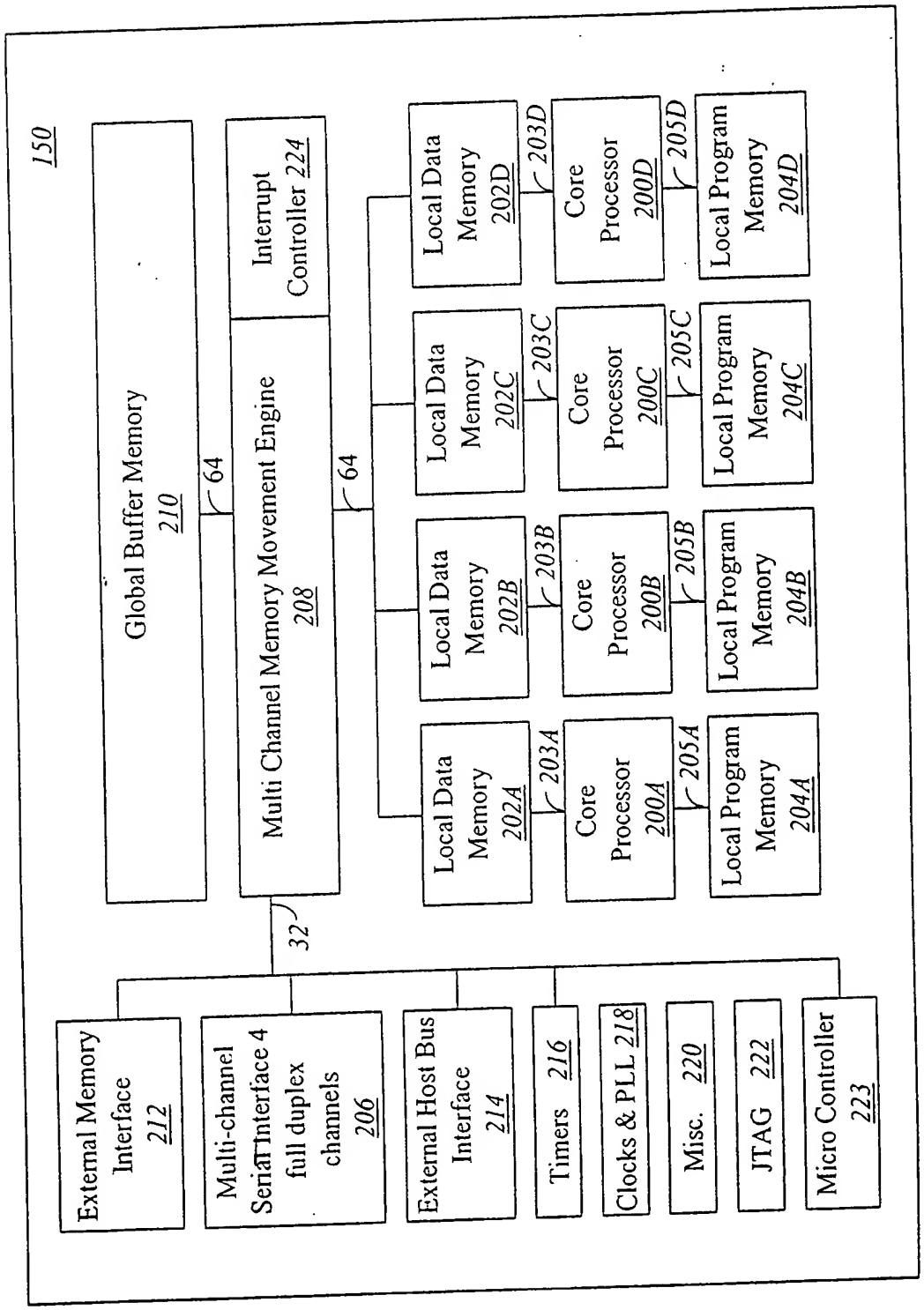

150
Global Buffer Memory 210
64
Interrupt Controller 224
Multi Channel Memory Movement Engine 208
64
32
Local Data Memory 202A
203A
Core Processor 200A
205A
Local Program Memory 204A
Local Data Memory 202B
203B
Core Processor 200B
205B
Local Program Memory 204B
Local Data Memory 202C
203C
Core Processor 200C
205C
Local Program Memory 204C
Local Data Memory 202D
203D
Core Processor 200D
205D
Local Program Memory 204D
External Memory Interface 212
Multi-channel SerialTInterface 4 full duplex channels 206
External Host Bus Interface 214
Timers 216
Clocks & PLL 218
Misc. 220
JTAG 222
Micro Controller 223

FIG. 2

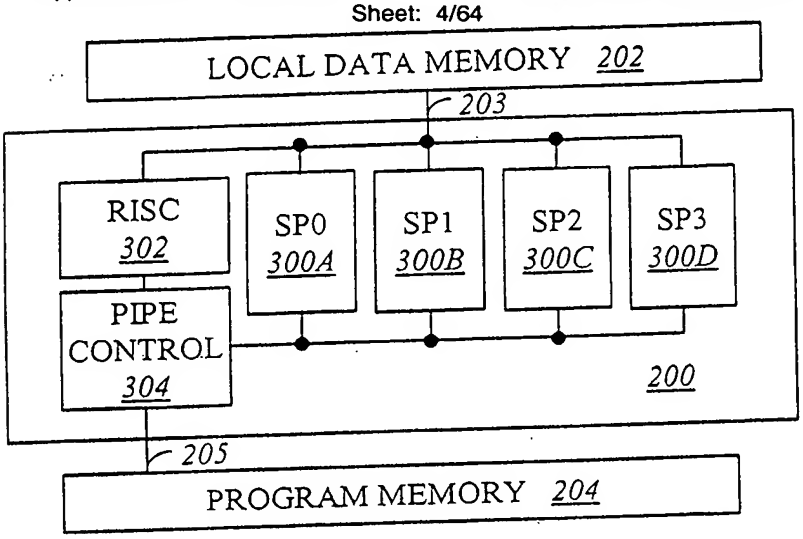


FIG. 3

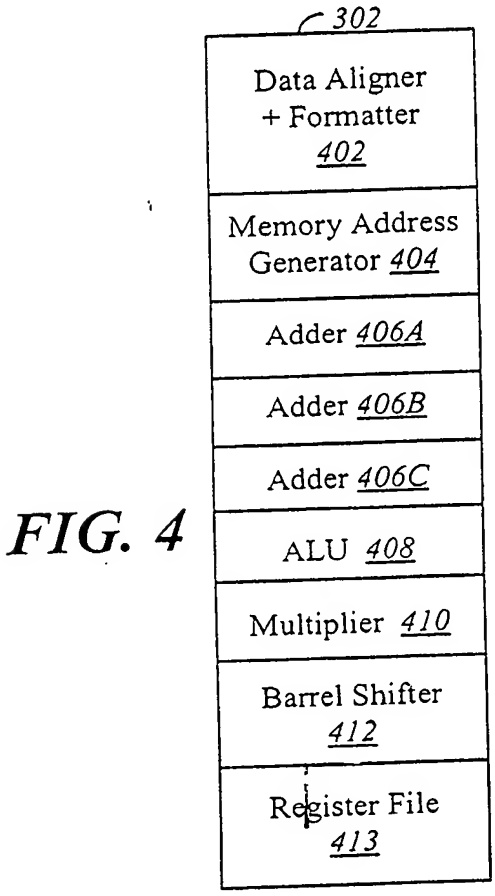


FIG. 4

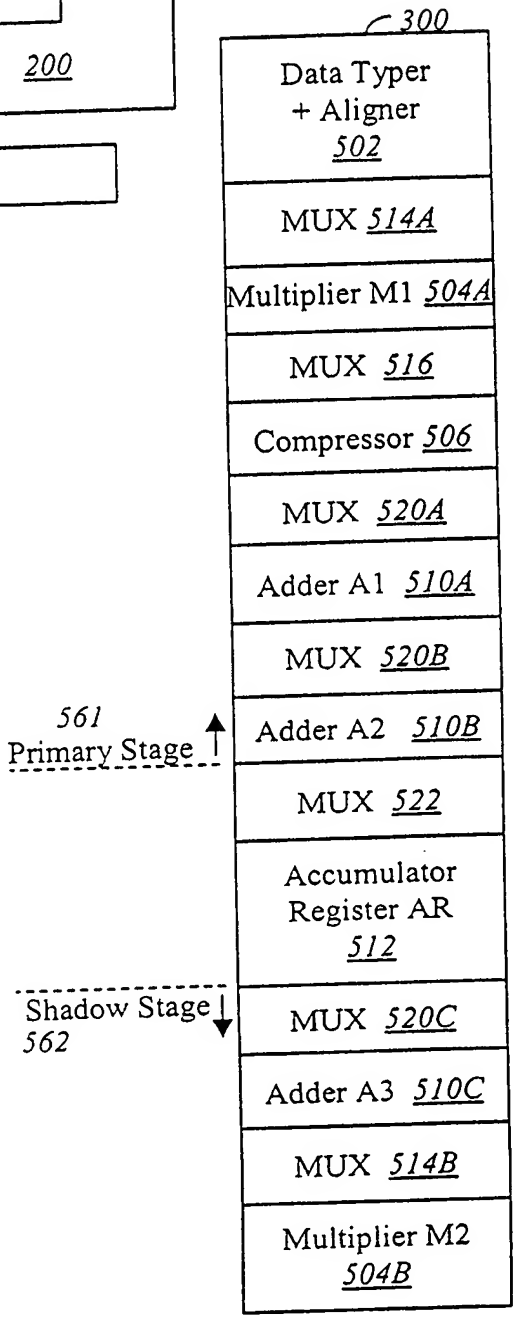


FIG. 5A

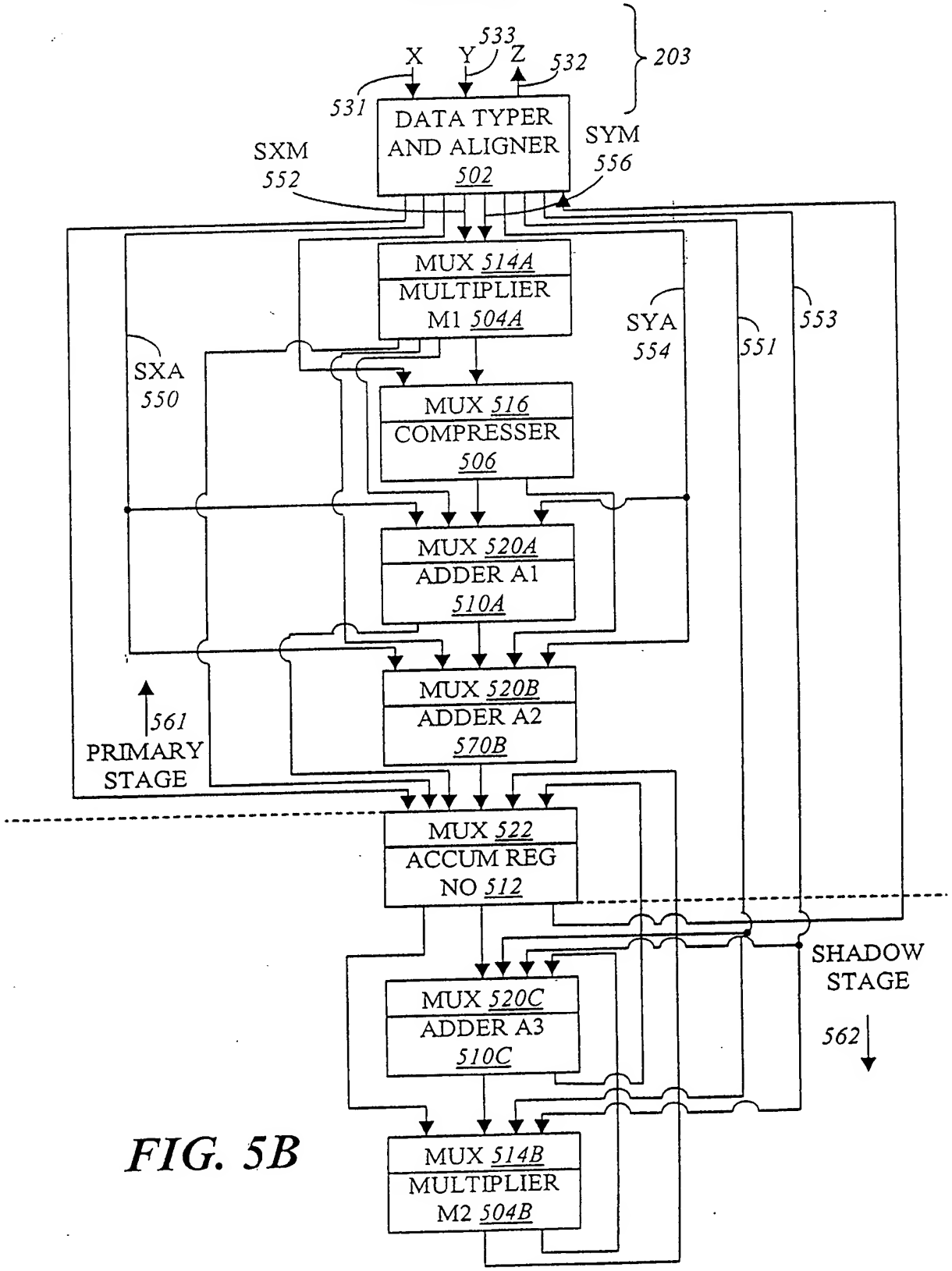

533
X
Y
Z
532
203
531
DATA TYPER AND ALIGNER 502
SXM 552
SYM 556
MUX 514A
MULTIPLIER M1 504A
SYA 554
551
553
SXA 550
MUX 516
COMPRESSER 506
MUX 520A
ADDER A1 510A
MUX 520B
ADDER A2 570B
561
PRIMARY STAGE
MUX 522
ACCUM REG NO 512
SHADOW STAGE
562
MUX 520C
ADDER A3 510C
MUX 514B
MULTIPLIER M2 504B

FIG. 5B

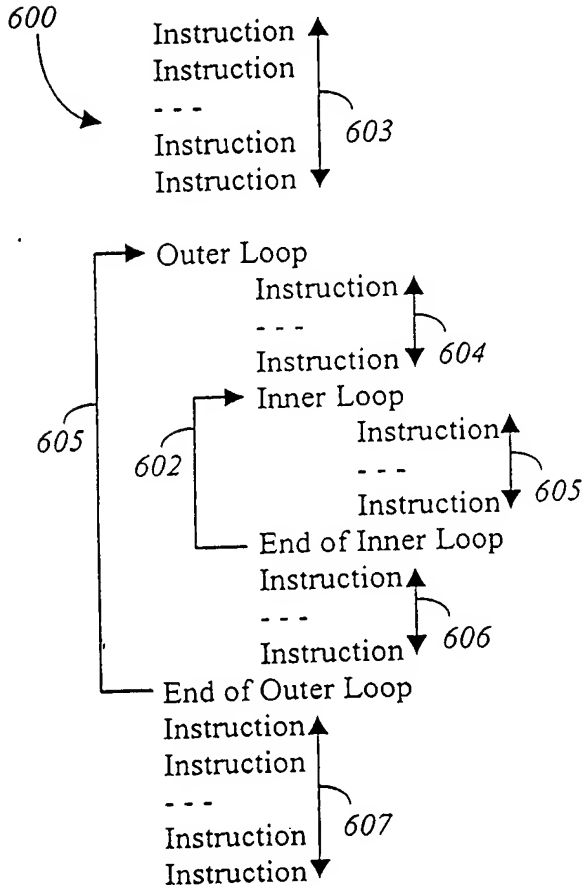


FIG. 6A

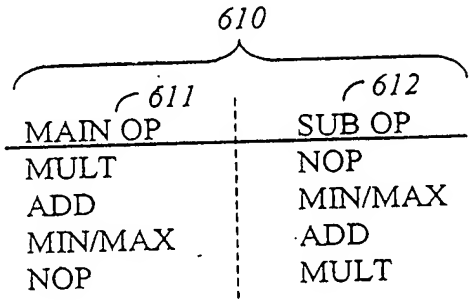


FIG. 6G

20-bit ISA

39	19	
0	0	20-bit parallel
0	1	20-bit serial
1	0	40-bit extended
1	1	20-bit serial

Control II Control
Control # Control
DSP, extensions/Shadow
DSP # DSP

FIG. 6B

6-bit operand specifier:

A 6-bit specifier is used in DSP extended instructions to access memory and register operands.

5	4	3	2	1	0

M/R

0	0	ac-page	ereg
0	1	gpr:0-r15	GPR
1	ptr(r0)to(r15)	off	Mem[ptr{0-15}]II PTR[0-15]+=offset1/offset2 Always postupdate

This allows access to data memory, ereg and GPR

• Bit 5=1:Use rX(X:0-7) register to obtain effective memory address and post-modify the
ptr field by one of two possible offsets specified in rX registers.
dmem[ptr], ptr=ptr+offset1, if off=0
ptr=ptr+offset2, if off=1

• Bit 5=0:Access ac-page or GPR

If Bit-4 is set to 0, then bits 3:0 control access to the general-purpose register file (r0-15) or to execution unit registers.

GPR
R0
R1
R2
R3
R4
R5
R6
R7
R8
R9
R10
R11
R12
R13
R14
R15

GPR Intr page
R0
R1
R2
R3
R4
R5
R6
R7
R8
R9
R10
R11
R12_i
R13_i
R14_i
R15_i

ac-page
A0
A1
T
TR

ac intr page
A0_I
A1_I
T
TR

ereg-Shadow DSP
A0
A1
T
TR
SX1
SX1s
SX2
SX2s
SY1
SY1s
SY2
SY2s

FIG. 6C

4-bit operand specifier:
Memory operands: (rX) specifies an access out of the data memory to the extension unit for the function that needs to be performed. The address for the access is specified in the rX register in the general register file that hold the 14-bit pointer (16K of addressing) to memory, 5-bit signed offset or a 3-bit unsigned offset that can post-modify the address. In addition, each pointer is typed for efficient SIMD processing and includes a permute control for rearranging data elements of a vector on the fly. The "podi" core can deal with 4-element 16-bit real vectors or complex data directly. This ability to manipulate memory data directly reduces the instruction width greatly and allows efficient signal processing.

(rX): Memory Address Registers

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
type				cb	x	permute				off1:(0-7)			off0:(-16 to 15)					ptr: pointer													

Fig. 6D

For shadow DSP instructions, the 3-bit specifier for operands is defined as follows:

2	1	0	
0	0	0	A0
0	0	1	A1
0	1	0	T
0	1	1	TR
1	0	0	SX1
1	0	1	SX1s
1	1	0	SX2
1	1	1	SX2s

EREG1

2	1	0	
0	0	0	A0
0	0	1	A1
0	1	0	T
0	1	1	TR
1	0	0	SY1
1	0	1	SY1s
1	1	0	SY2
1	1	1	SY2s

EREG2

Only the shadow DSP instructions can see the above modified page of execution unit registers.

FIG. 6E

5-bit operand specifier:

The 5-bit specifier includes the 4-bit specifier for general data operands and the special purpose registers. It is used in RISC instructions.

4	3	2	1	0
0	spr:s0-s15			
1	gpr:r0-r15			

	SPR SPR	Intr page SPR intr page	
0	fu-ctl	fu-ctl_I	
1	a-type	a-type_I	
2	ps-ctl	ps-ctl	
3	t-type	t-type	
4	pl-ctl	pl-ctl	
5	cb-ctl	cb-ctl_I	
6	shuffle	shuffle	
7	lo-ptr	lo-ptr	
8	status	status_I	
9	loop-ctl	loop-ctl	
10	pcr	pcr	stack(8)
11	reserved	reserved	
12	reserved	reserved	
13	reserved	reserved	
14	reserved	reserved	

NOTE: All SPR registers are reset to all zeros at power on reset except for the PCR register.

FIG. 6F

DSP instructions

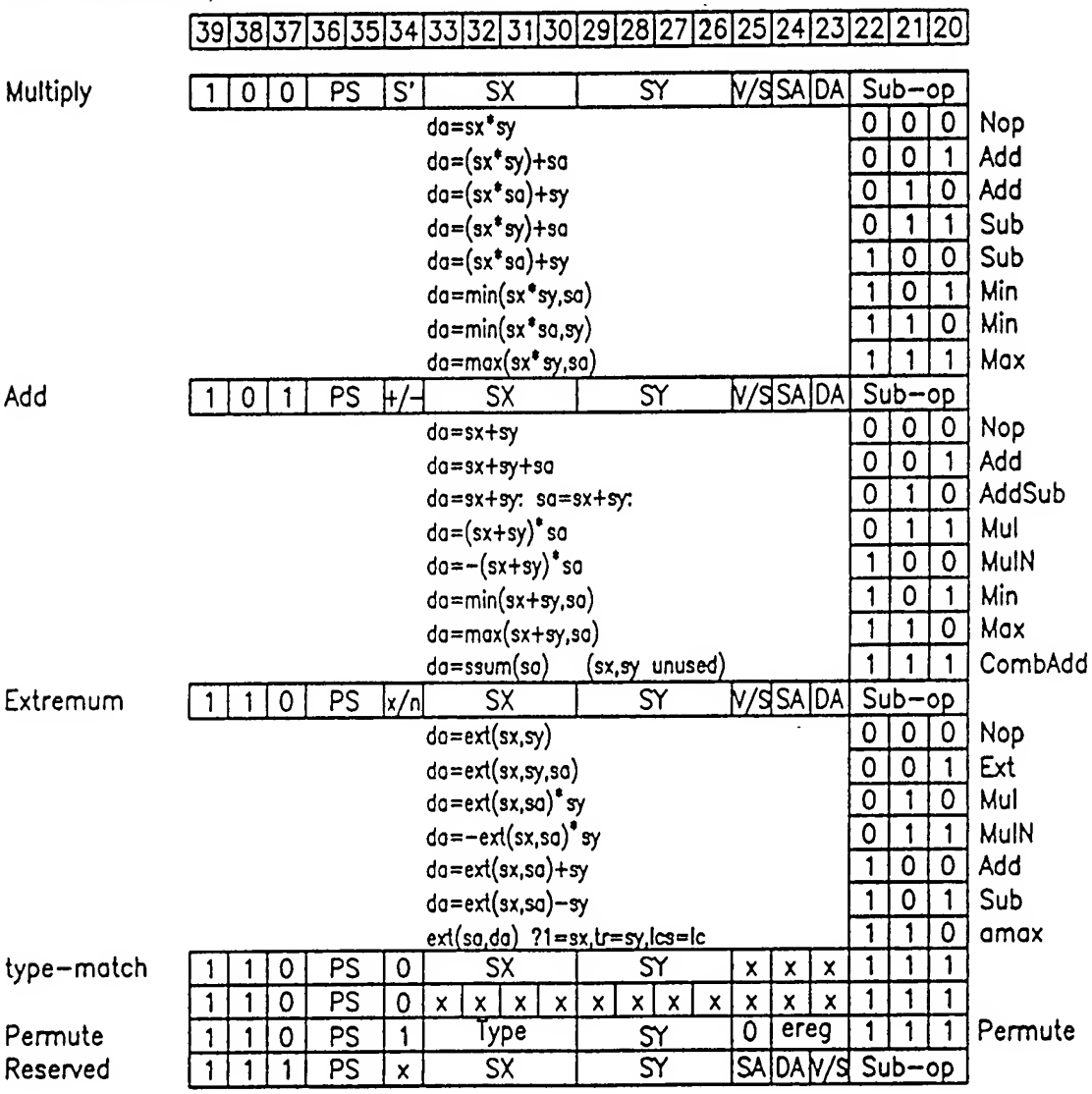
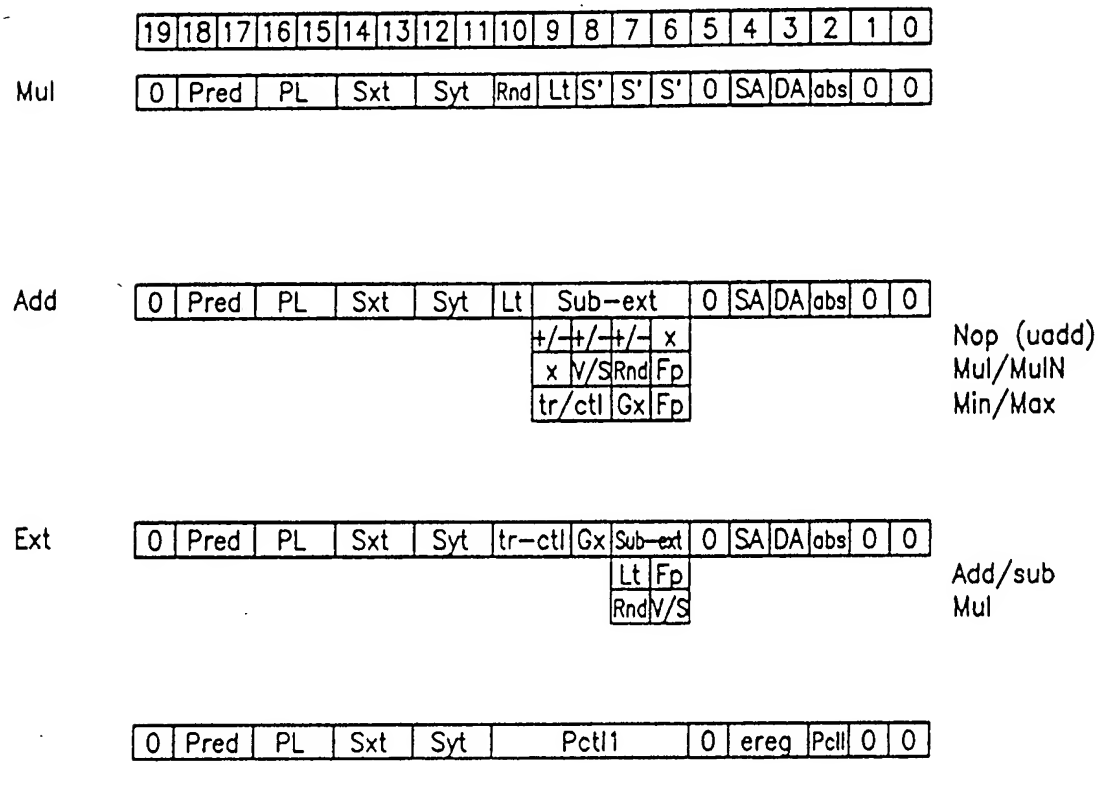
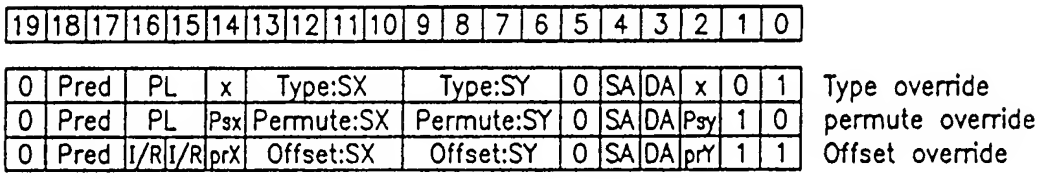


FIG. 6H

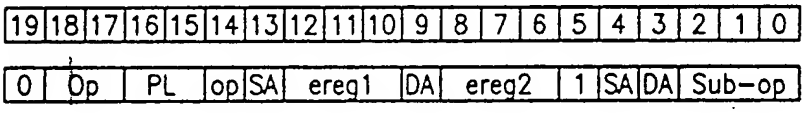
Control and specifier Extensions



Type/offset/permute extensions



Shadow DSP



nop

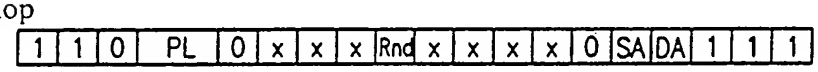


FIG. 6I

Control instructions

	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
add,sub	L	Pred		0	0	0	RX				RY				RZ				+/-	0	
max,min	L	Pred		0	0	0	RX				RY				RZ				X/N	1	
Shift	L	Pred		0	0	1	RX				UI4				RZ				UI2	R/L	<Bit1,Bits9–6>
Logic	L	Pred		0	1	0	RX				RY				RZ				&,	&1	==UI5 (Shift
Mux	L	Pred		0	1	1	RX				RY				RZ				Pd	0	Amount)
mov	L	Pred		0	1	1	RX				DZ				Rxt	Dzt	0	0	0	1	
addi	L	Pred		0	1	1	SI4				DZ				x	x	1	0	0	1	
mov2erg	L	Pred		0	1	1	RX				unit		ereg		qd		type		1	0	1
l.dm	L	Pred		0	1	1	RX				DZ1				DZ2				1	1	
Set4bits	L	Pred		1	0	0	UI4:POS				RZ				Rzt		UI4			0	
Set2bits	L	Pred		1	0	0	UI4:POS				RZ				Rzt		UI2		0	0	1
Setbit	L	Pred		1	0	0	UI4:POS				RZ				Rzt	UI1	UI1	1	0	1	<Bit3,Bits13–
Movi	L	Pred		1	0	0	SI8								RZ				1	1	10>==UI5 POS
Jmp	L	Pred		1	0	1	SI9								0	PRED			0	0	
Call	L	Pred		1	0	1	SI9								1	PRED			0	0	
Loop	L	Pred		1	0	1	UI5:Lcount					UI5:Lsize					UI2:Lst		0	1	
Jmpi	L	Pred		1	0	1	RX				x	x	x	x	x	0	PRED		1	0	
Calli	L	Pred		1	0	1	RX				x	x	x	x	x	1	PRED		1	0	
Loopi	L	Pred		1	0	1	RX				x	UI5:Lsize					UI2:Lst		1	1	
Test	L	Pred		1	1	0	RX				RY				PZ			=,<,>		0	
Testbit	L	Pred		1	1	0	RX				UI5				PZ		B		0	1	
Andp, orp	L	Pred		1	1	0	Pa				Pb			Pc	PZ			&	1	1	
Load	L	Pred		1	1	1	MX				RZ				Ext			0	0	0	
Store	L	Pred		1	1	1	MZ				RZ				Ext			1	0	0	
eLoad	L	Pred		1	1	1	MX				RY				1	1	1	0	0	0	
eStore	L	Pred		1	1	1	MZ				RY				1	1	1	1	0	0	
Extended	L	Pred		1	1	1	Bits 27:16												1	0	
Logic2	L	Pred		1	1	1	RX				RY/RZ				Rxt	Ryt	&,1,&1,!		0	1	
mov–erg	L	Pred		1	1	1	unit		ereg		RZ				qd		Sft		0	1	1
Crb	L	Pred		1	1	1	RX				RZ				s/m	0	0	1	1	1	
Panty	L	Pred		1	1	1	RX				PZ			0/E	0	1	0	1	1	1	
Stm	L	Pred		1	1	1	MZ				RX				1	1	0	1	1	1	
Abs	L	Pred		1	1	1	RX				RZ				0	0	1	1	1	1	
Neg	L	Pred		1	1	1	RX				RZ				0	1	1	1	1	1	
Div–step	L	Pred		1	1	1	RX				RZ				1	0	1	1	1	1	
Test&Set	L	Pred		1	1	1	RX				PZ			0	1	1	1	1	1	1	
Reserved	L	Pred		1	1	1						0	0	1	1	1	1	1	1	1	
Return	L	Pred		1	1	1	Pred		I–ctl		0	1	0	1	1	1	1	1	1	1	
Zero–ac	L	Pred		1	1	1	ac #				1	1	0	1	1	1	1	1	1	1	
eSync	L	Pred		1	1	1	RZ				0	1	1	1	1	1	1	1	1	1	
Swi	L	Pred		1	1	1	UI3			0	1	1	1	1	1	1	1	1	1	1	
Nop	L	Pred		1	1	1	UI3			1	1	1	1	1	1	1	1	1	1	1	

FIG. 6J

Extended Control

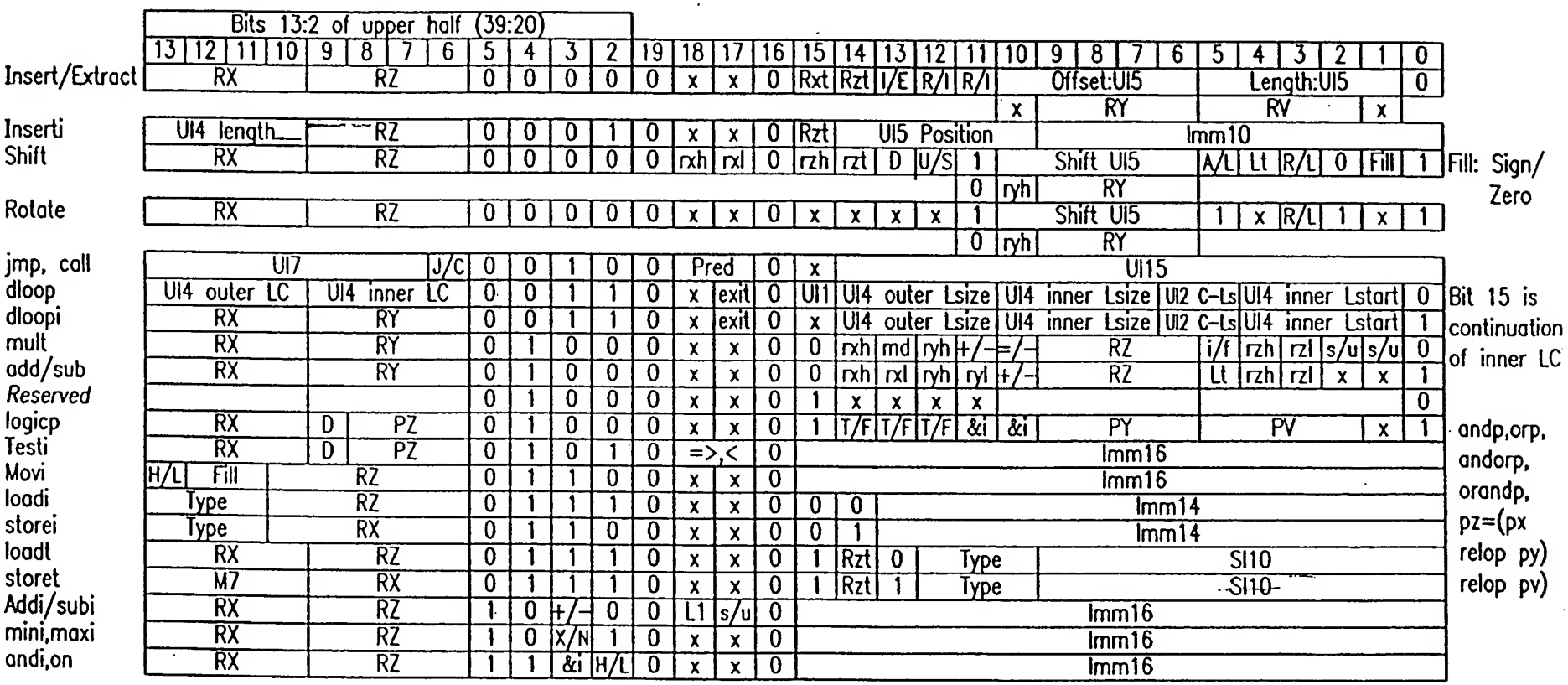


FIG. 6K

REPLACEMENT SHEET
Title: METHOD AND APPARATUS OF BUS STATE KEEPERS FOR BUSES IN AN INTEGRATED CIRCUIT
1st Named Inventor: Ruban Kanapathippillai
Application No.: 10/649,067 Docket No.: 42P14037D2
Sheet: 13/64

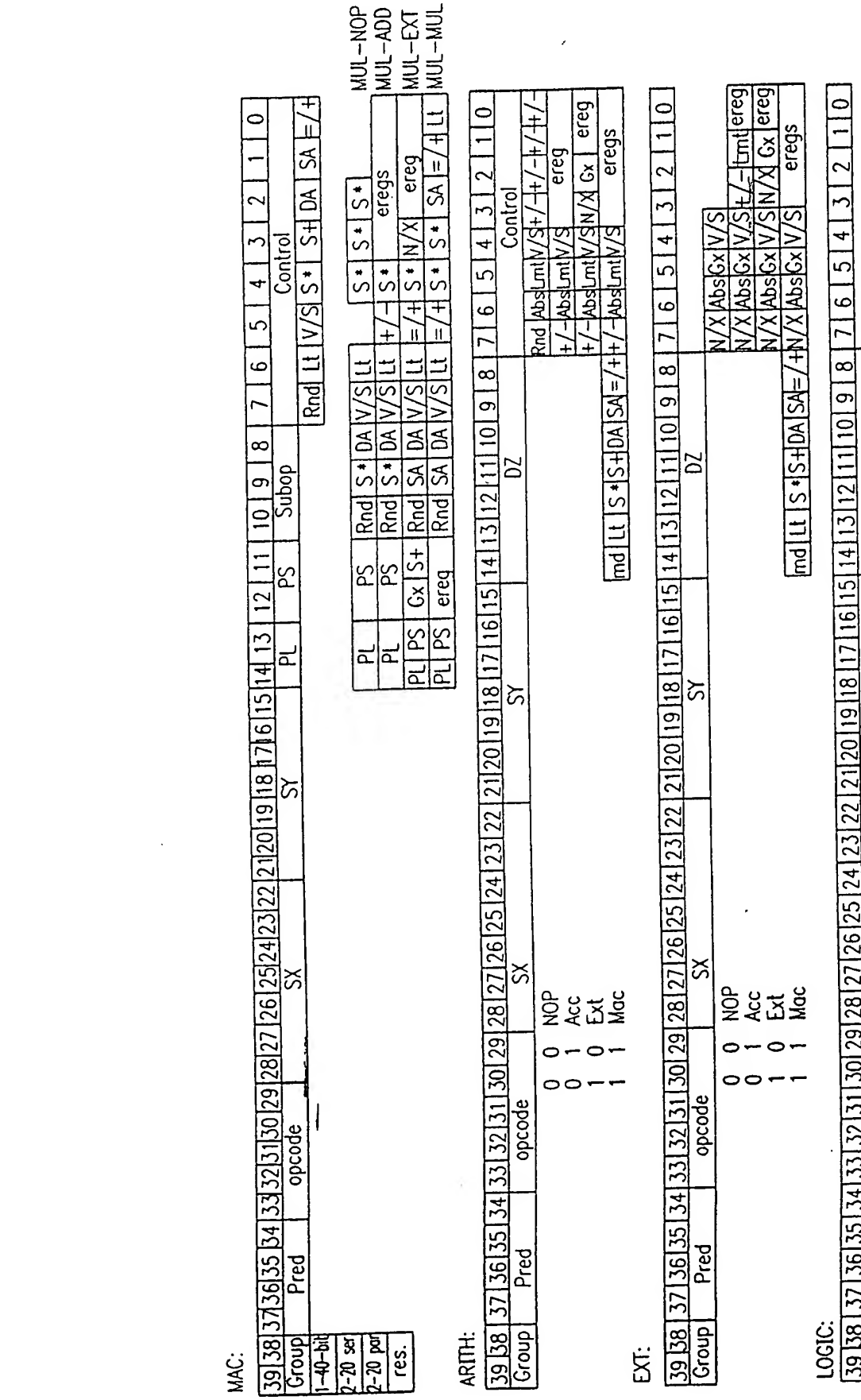

MAC:
ARITH:
EXT:
LOGIC:
Group
Pred
opcode
SX
SY
DZ
Subop
Control
eregs
1-40-bit
2-20 ser
2-20 par
res.
MUL-NOP
MUL-ADD
MUL-EXT
MUL-MUL
NOP
Acc
Ext
Mac

FIG. 6L-1

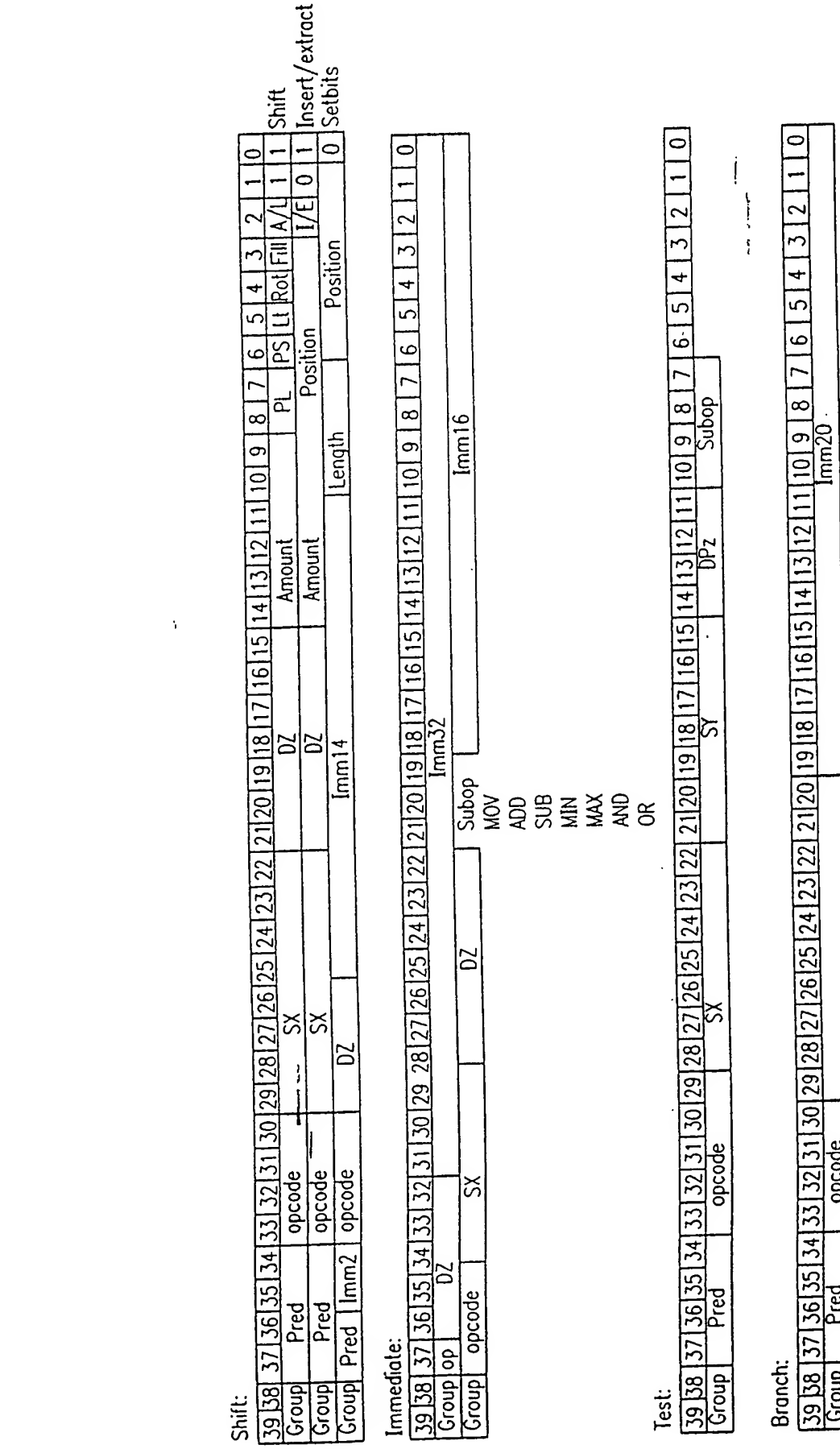

Shift:
Group Pred opcode SX Amount DZ PL PS Lt Rot Fill A/L Shift
Group Pred opcode SX Amount DZ Position I/E 0 Insert/extract
Group Pred Imm2 opcode DZ Imm14 Length Position 0 Setbits
Immediate:
Group op DZ Imm32
Group opcode SX DZ Subop Imm16
MOV
ADD
SUB
MIN
MAX
AND
OR
Test:
Group Pred opcode SX SY DPz Subop
Branch:
Group Pred opcode Imm20
Misc:

FIG. 6L-2

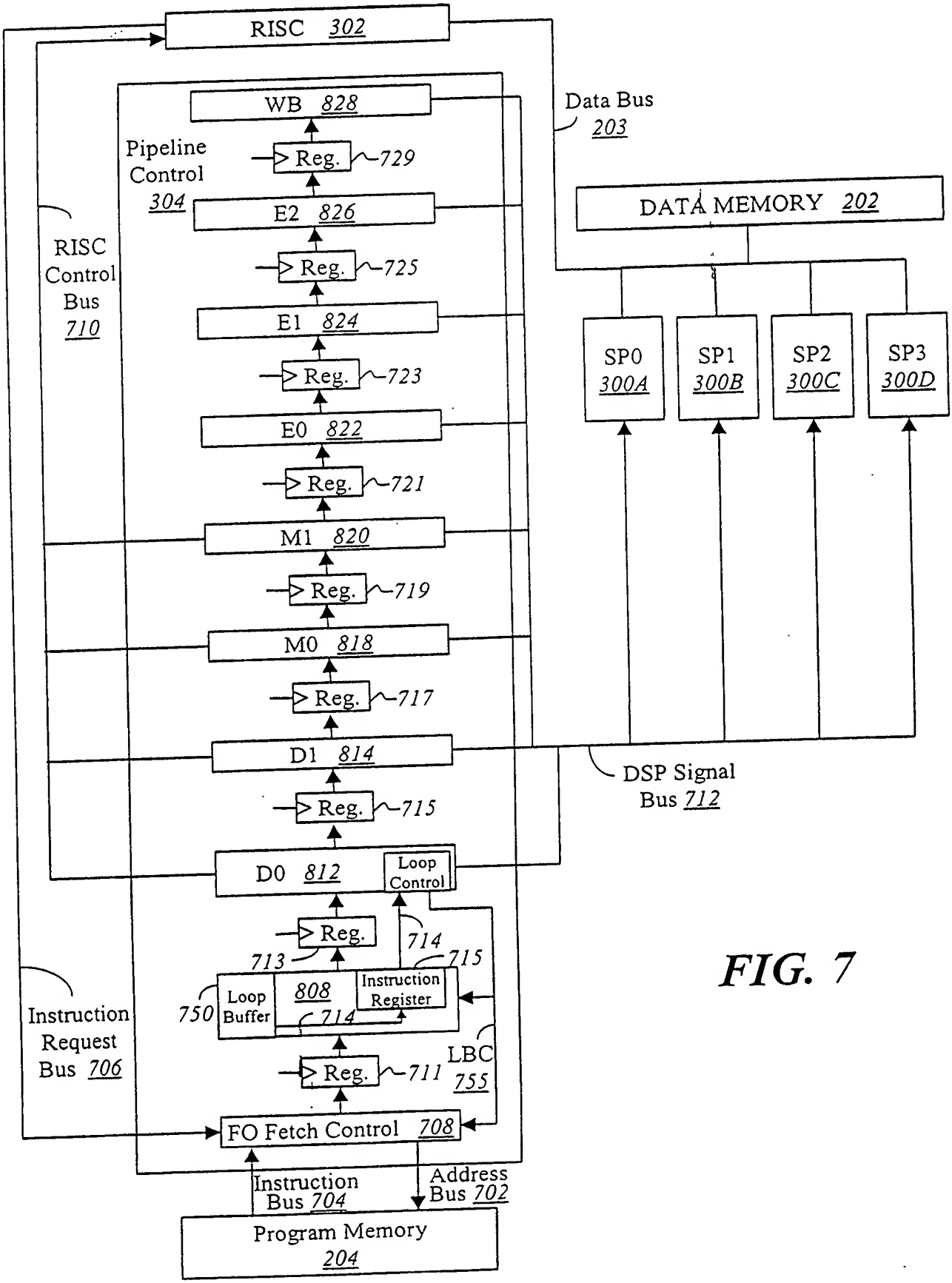

RISC 302
WB 828
Pipeline Control 304
Reg. 729
E2 826
Reg. 725
E1 824
Reg. 723
E0 822
Reg. 721
M1 820
Reg. 719
M0 818
Reg. 717
D1 814
Reg. 715
D0 812
Loop Control
Reg.
713
714
715
Loop Buffer
808
Instruction Register
750
714
Reg. 711
LBC 755
FO Fetch Control 708
Instruction Bus 704
Address Bus 702
Program Memory 204
RISC Control Bus 710
Instruction Request Bus 706
Data Bus 203
DATA MEMORY 202
SP0 300A
SP1 300B
SP2 300C
SP3 300D
DSP Signal Bus 712

FIG. 7

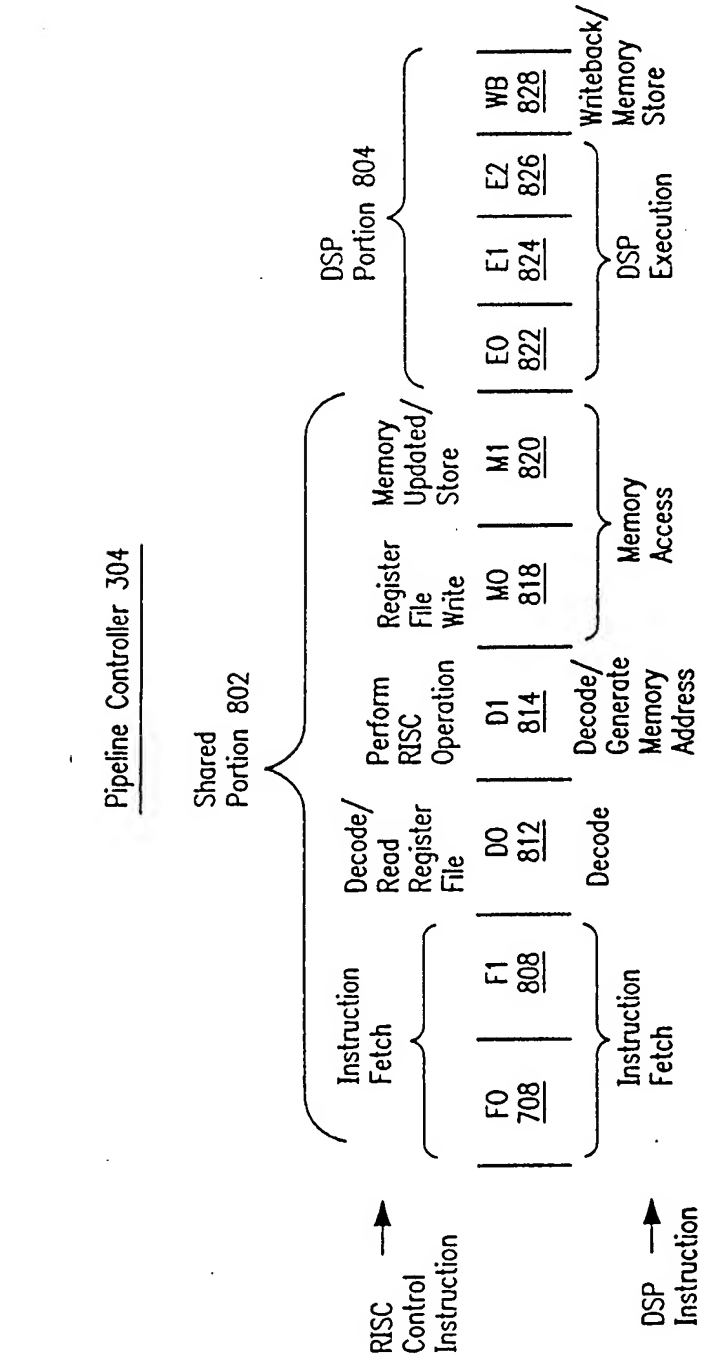

Pipeline Controller 304
Shared Portion 802
DSP Portion 804
RISC Control Instruction
DSP Instruction
Instruction Fetch
Decode/ Read Register File
Perform RISC Operation
Register File Write
Memory Updated/ Store
F0 708
F1 808
D0 812
D1 814
M0 818
M1 820
E0 822
E1 824
E2 826
WB 828
Decode
Decode/ Generate Memory Address
Memory Access
DSP Execution
Writeback/ Memory Store

FIG. 8a

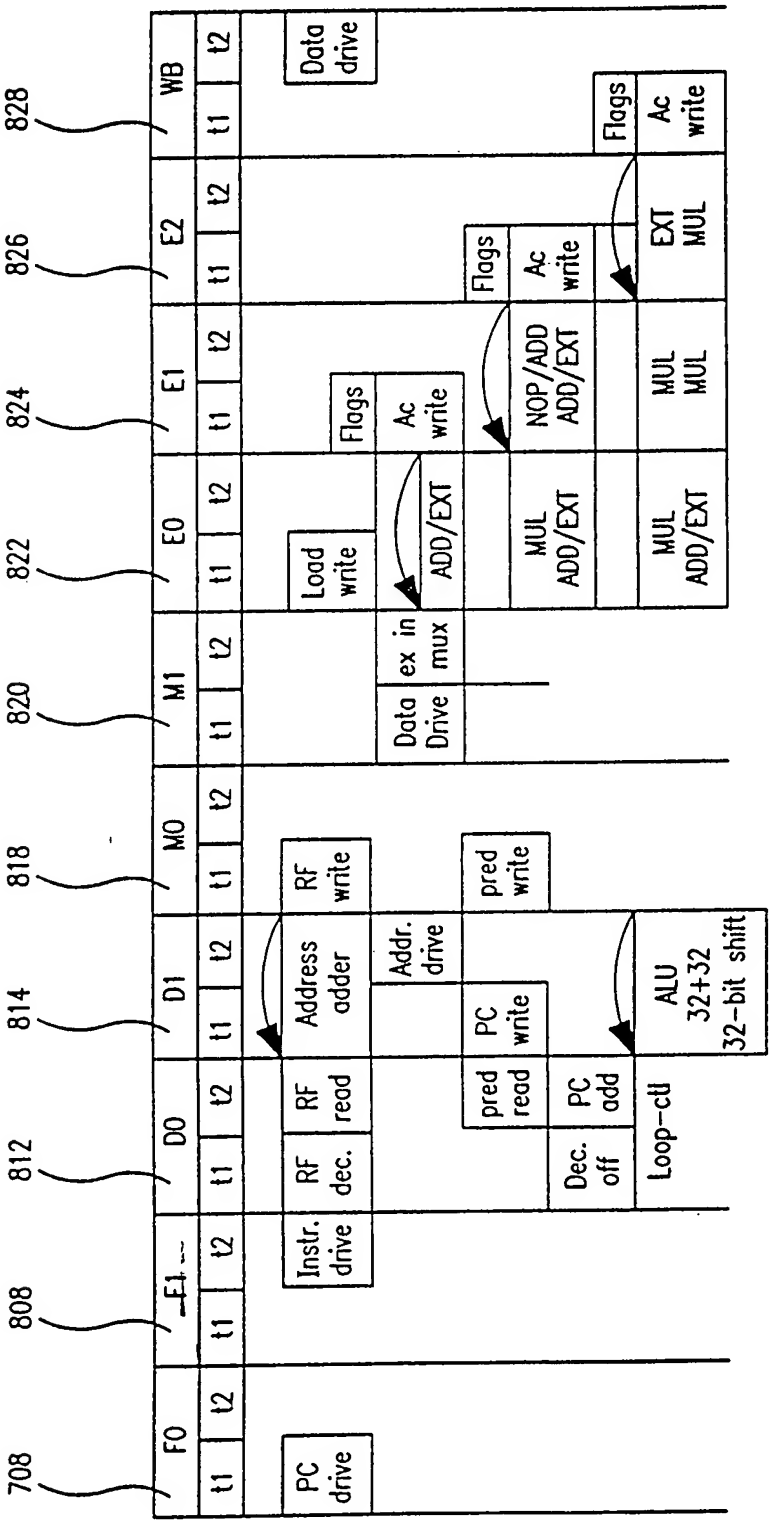

708
808
812
814
818
820
822
824
826
828
F0
F1
D0
D1
M0
M1
E0
E1
E2
WB
t1
t2
PC drive
Instr. drive
RF dec.
RF read
Address adder
RF write
Addr. drive
ex in mux
Data Drive
Load write
ADD/EXT
MUL ADD/EXT
Flags
Ac write
NOP/ADD ADD/EXT
MUL MUL
Flags
Ac write
EXT MUL
Data drive
Flags
Ac write
PC write
pred read
pred write
PC add
Dec. off
Loop-ctl
ALU 32+32 32-bit shift

FIG. 8b

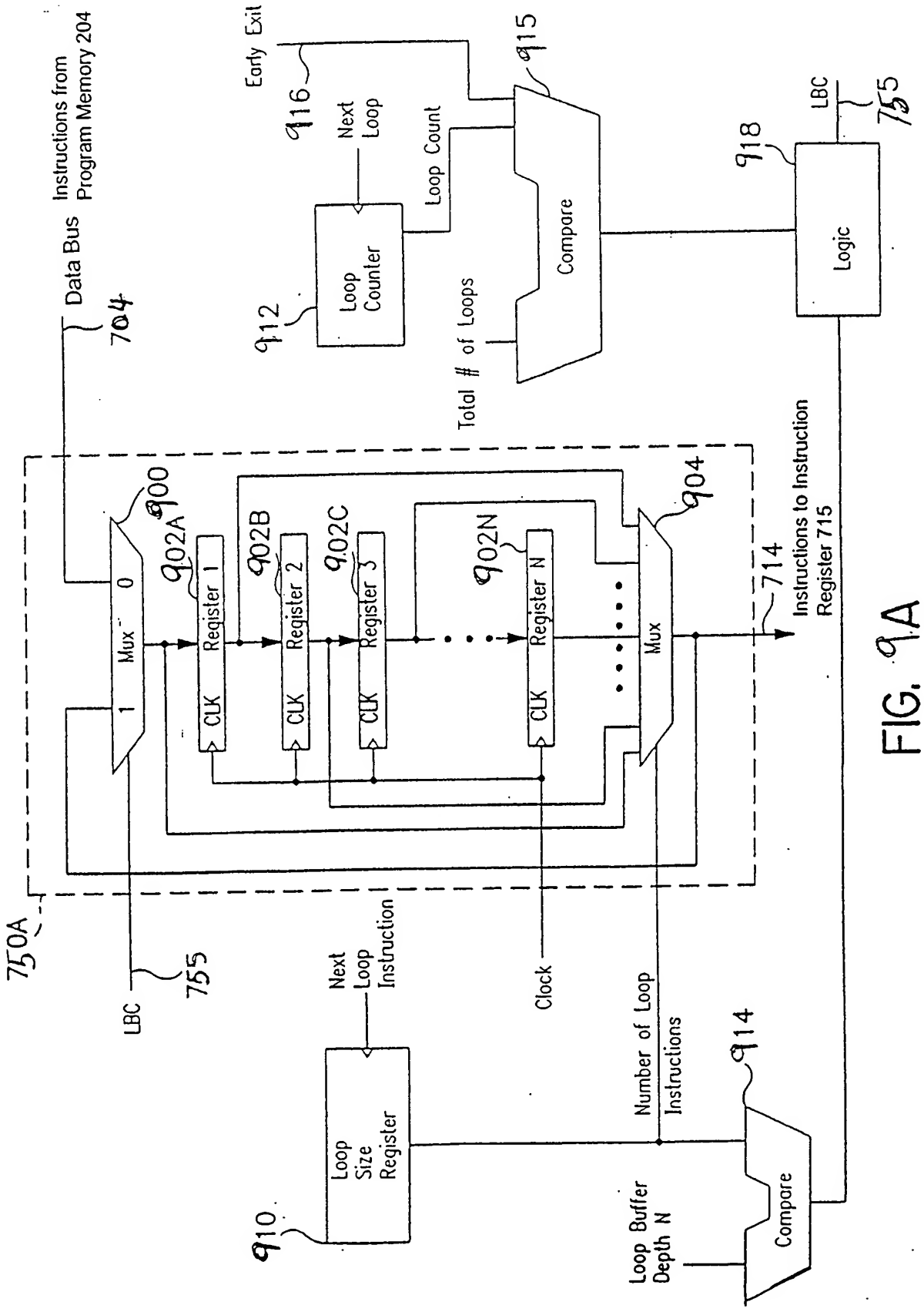

Instructions from Program Memory 204
Data Bus
704
Early Exit
916
Next Loop
Loop Count
915
Loop Counter
912
Total # of Loops
Compare
918
Logic
LBC
755
900
902A
902B
902C
902N
Register 1
Register 2
Register 3
Register N
CLK
Mux
904
714
Instructions to Instruction Register 715
750A
LBC
755
Next Loop Instruction
Clock
Number of Loop Instructions
914
Loop Size Register
910
Loop Buffer Depth N
Compare

FIG. 9A

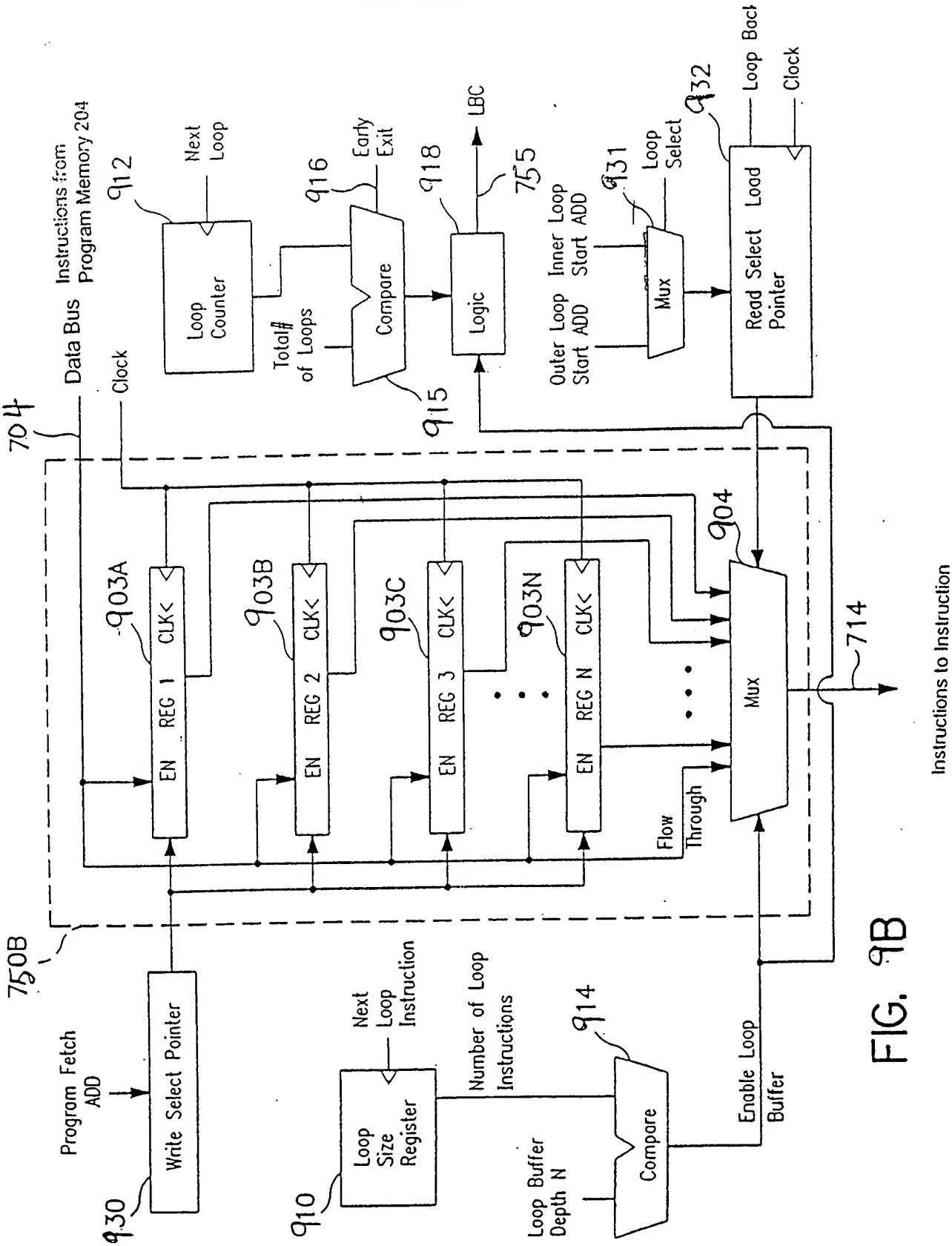

Instructions from Program Memory 204
Data Bus
Clock
704
912
Next Loop
Loop Counter
916
Early Exit
Total# of Loops
Compare
915
918
LBC
Logic
755
931
Loop Select
Outer Loop Start ADD
Inner Loop Start ADD
Mux
932
Loop Back
Clock
Read Select Load Pointer
903A
903B
903C
903N
EN REG 1 CLK<
EN REG 2 CLK<
EN REG 3 CLK<
EN REG N CLK<
904
Mux
714
Flow Through
750B
Program Fetch ADD
Write Select Pointer
930
910
Loop Size Register
Next Loop Instruction
Number of Loop Instructions
914
Loop Buffer Depth N
Compare
Enable Loop Buffer
Instructions to Instruction Register 715

FIG. 9B

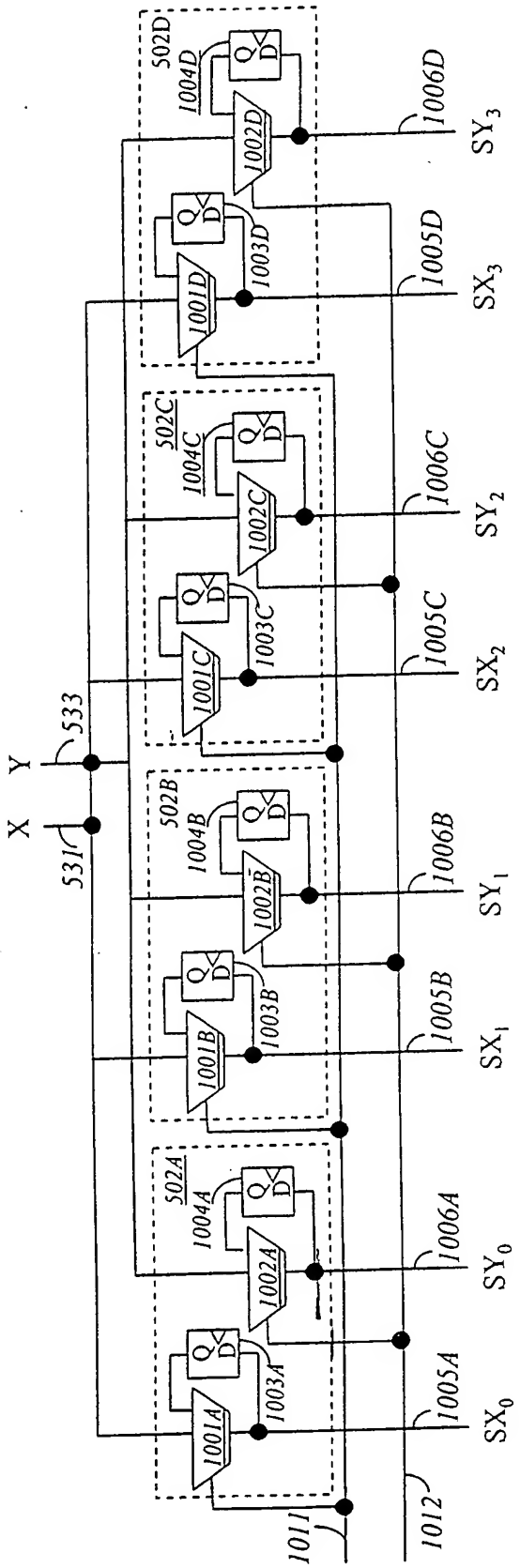

502D
1004D
1002D
1001D
1003D
1006D
1005D
SY3
SX3
502C
1004C
1002C
1001C
1003C
1006C
1005C
SY2
SX2
502B
1004B
1002B
1001B
1003B
1006B
1005B
SY1
SX1
502A
1004A
1002A
1001A
1003A
1006A
1005A
SY0
SX0
X
Y
531
533
1011
1012

FIG. 10

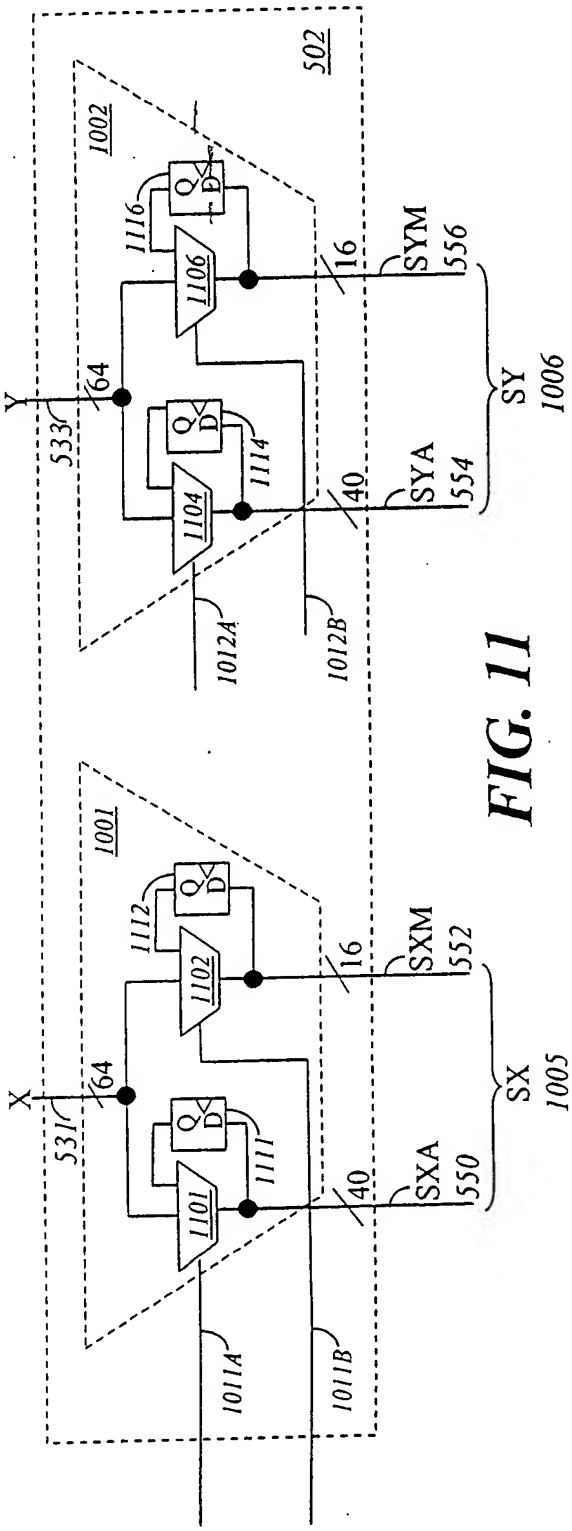
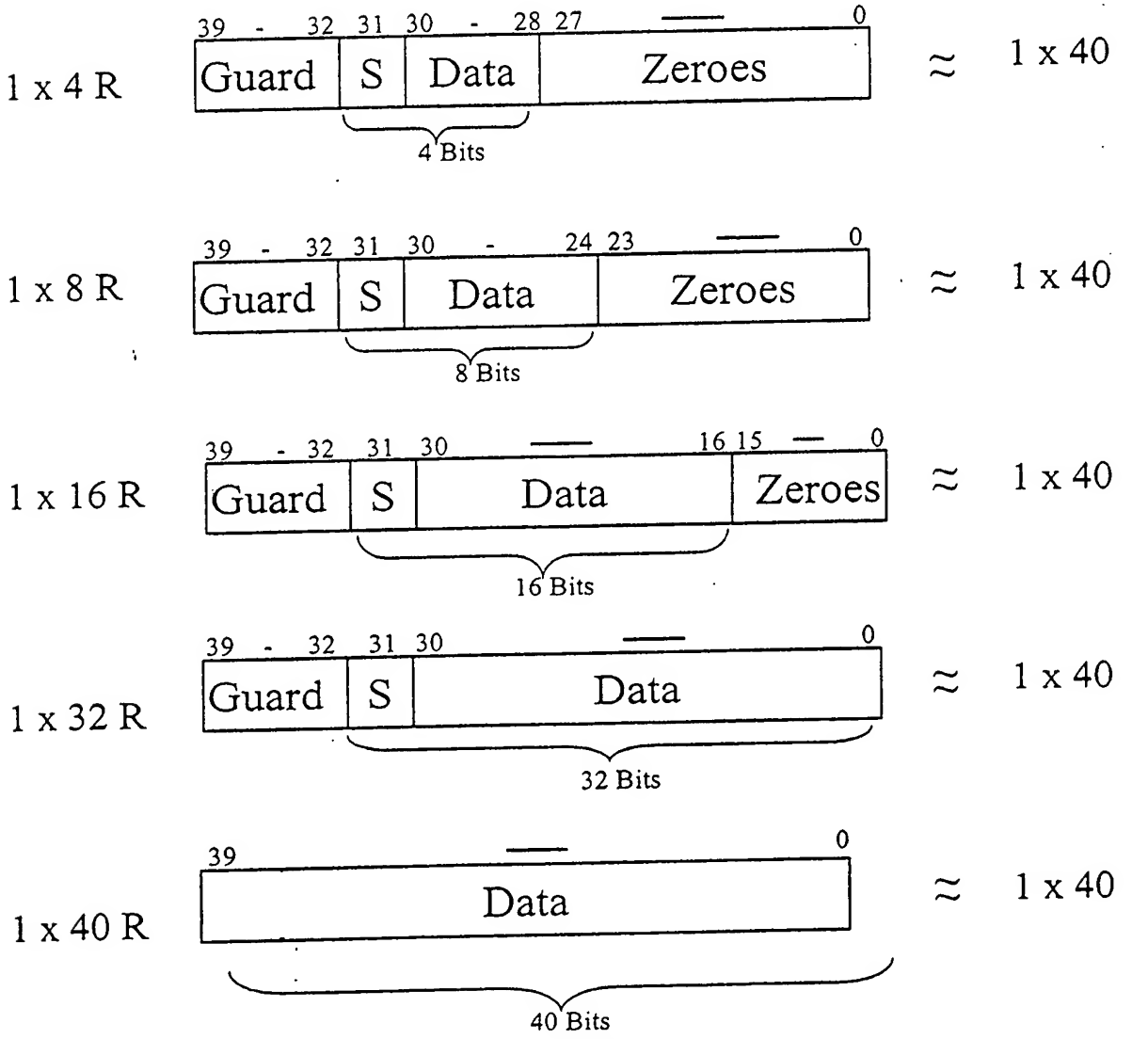

502
1002
1116
1106
1114
1104
1012A
1012B
Y
533
64
16
40
SYM
556
SYA
554
SY
1006
1001
1112
1102
1111
1101
1011A
1011B
X
531
64
16
40
SXM
552
SXA
550
SX
1005

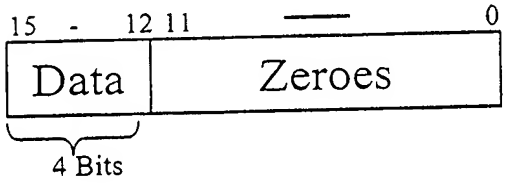
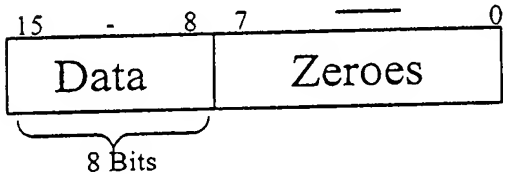
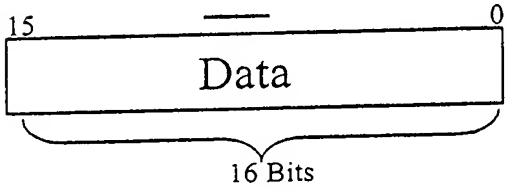
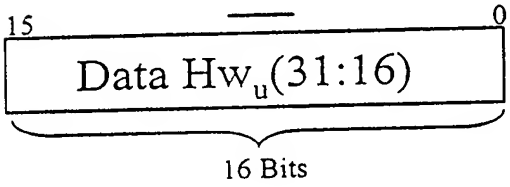
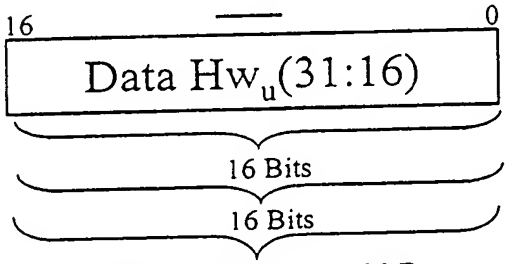
FIG. 11

FIG. 12A



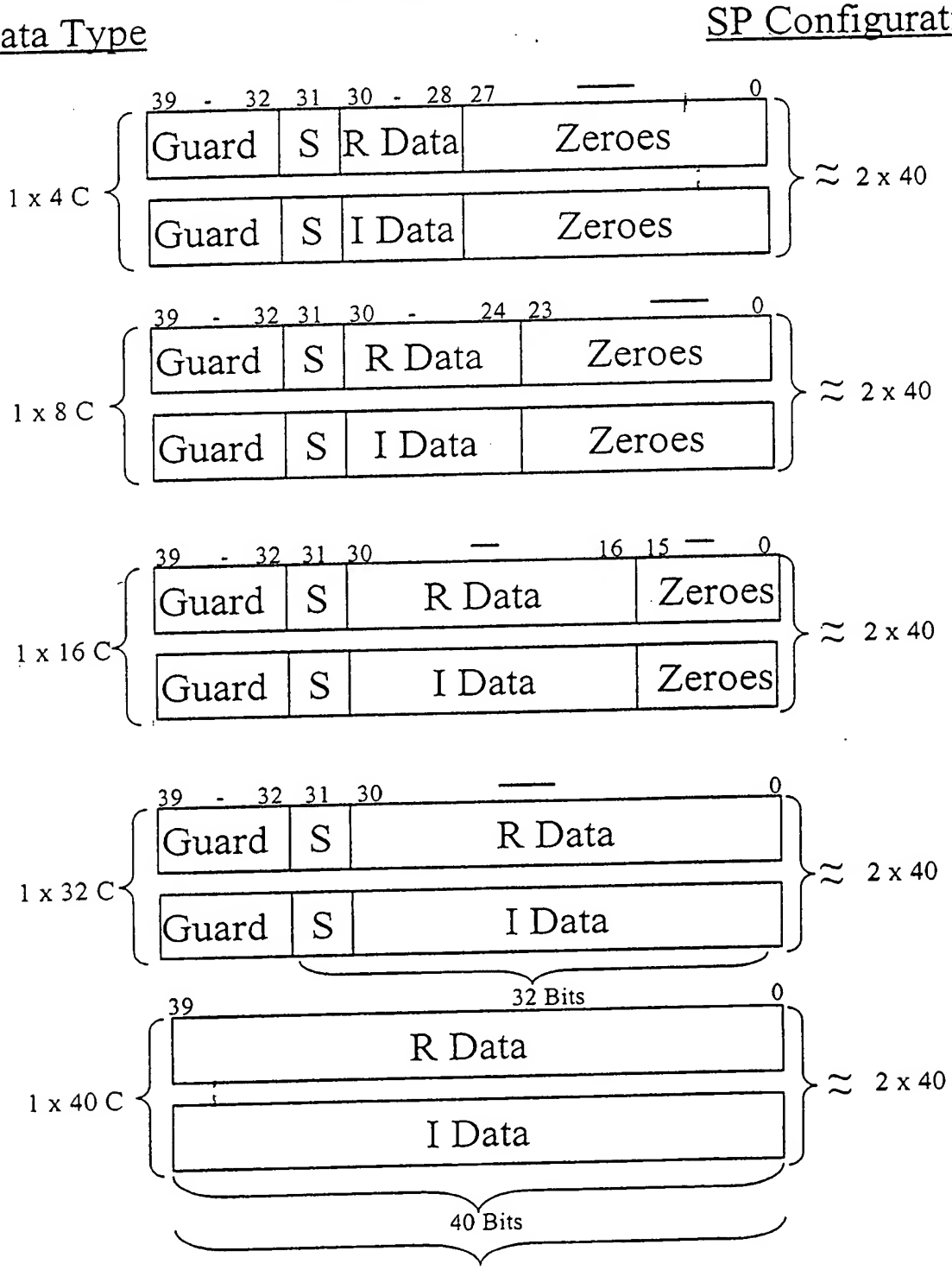
SXA *550* or SYA *554*

FIG. 12B

Data Type		SP Configuration
1 x 4 R		≈ 1 x 16
1 x 8 R		≈ 1 x 16
1 x 16 R		≈ 1 x 16
1 x 32 R		≈ 1 x 16
1 x 40 R		≈ 1 x 16

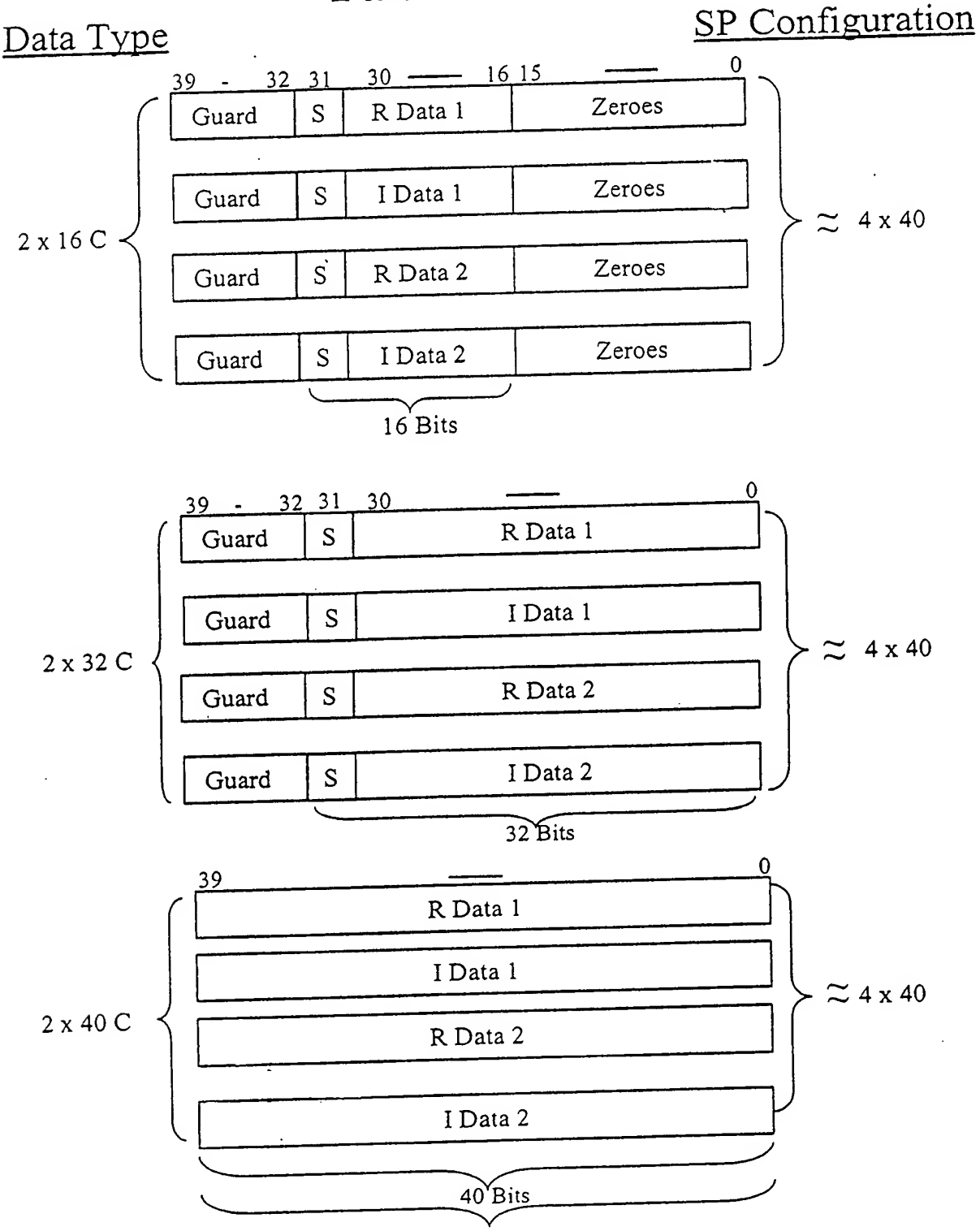
SXM *552A – 552B*
or
SYM *556A – 556B*

FIG. 12C



SXA *550A* and SXA *550B*
or
SYA *554A* and SYA *554B*

FIG. 12D

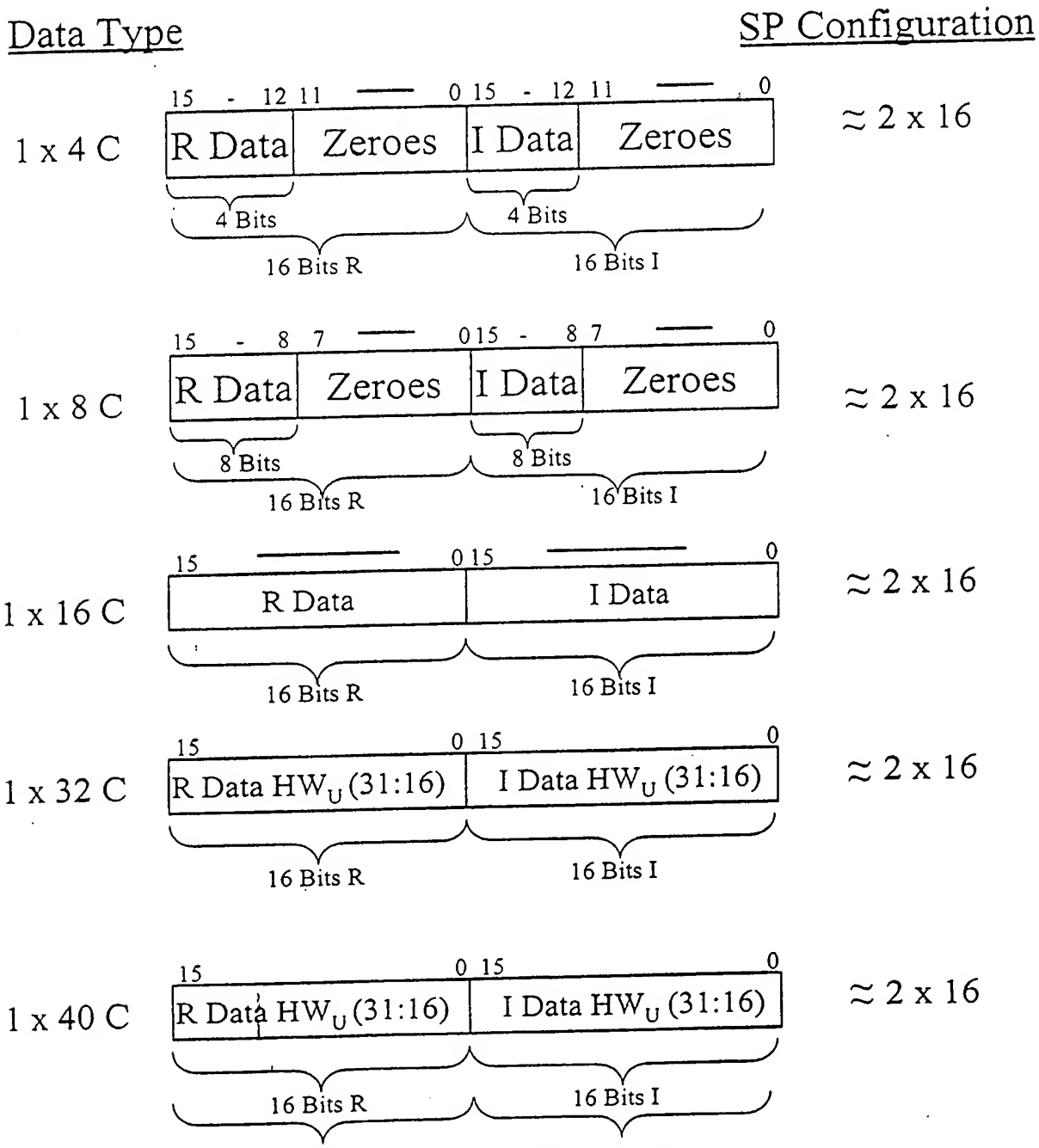


SXA *550A*, SXA *550B*, SXA *550C*, and SXA *550D*

or

SYA *554A*, SYA *554B*, SYA *554C*, and SYA *554D*

FIG. 12E

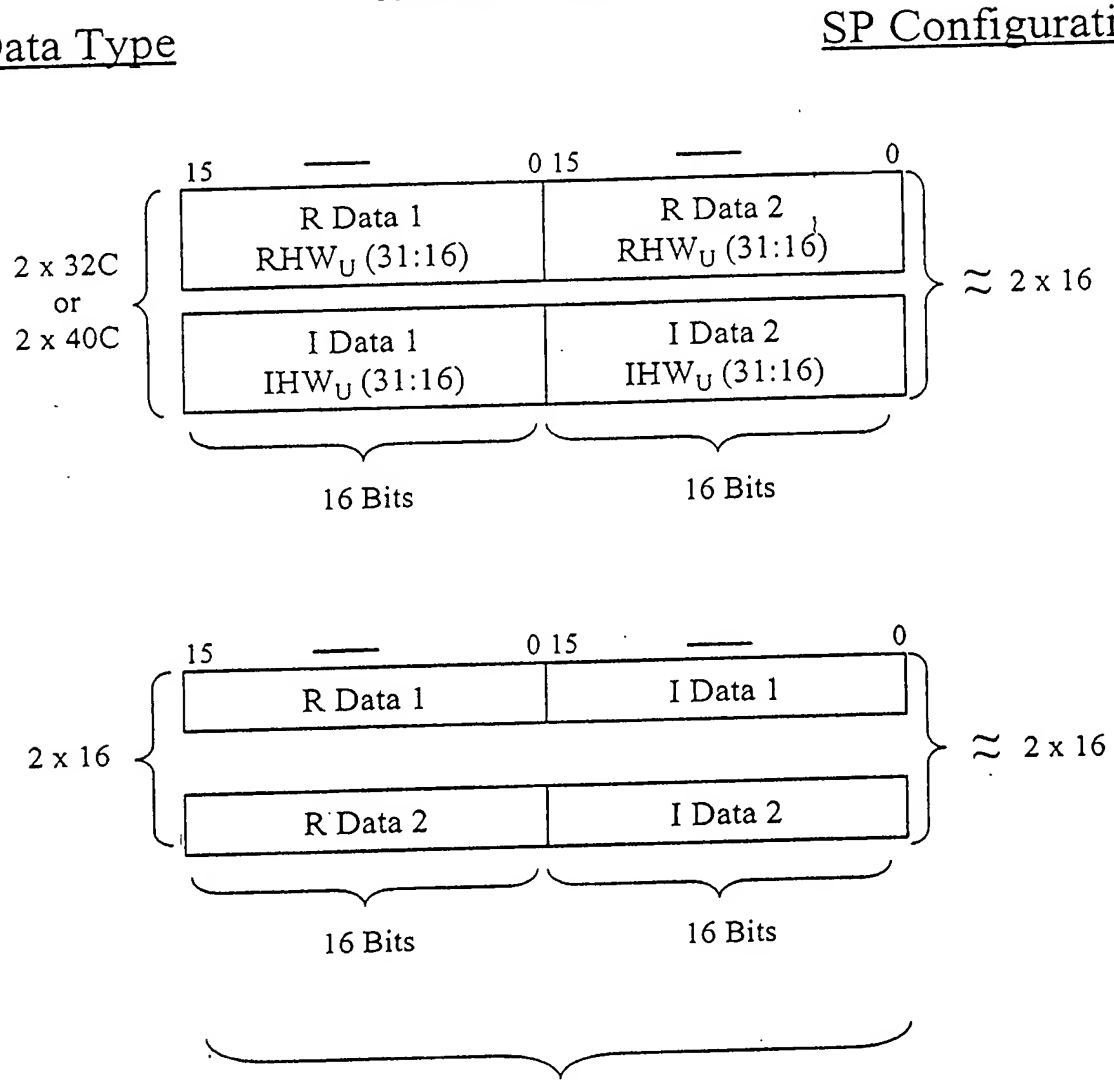


SXM *552A* and SXM *552B*

or

SYM *556A* and SYM *556B*

FIG. 12F



SXM *552A*, SXM *552B*, SXM *552C*, and SXM *552D*
or
SYM *556A*, SYM *556B*, SYM *556C*, and SYM *556D*

Operand 1 Data Type:	$N_1 \times S_1$ R
Operand 2 Data Type:	$N_2 \times S_2$ R
Type Matching R:	Max (N_1 or N_2) x Max (S_1 or S_2) R

Fig. 13A

Operand 1 Data Type:	$N_1 \times S_1$ C
Operand 2 Data Type:	$N_2 \times S_2$ C
Type Matching C:	Max (N_1 or N_2) x Max (S_1 or S_2) C

Fig. 13B

Operand 1 Data Type:	$N_1 \times S_1$ R
Operand 2 Data Type:	$N_2 \times S_2$ C
Type Matching R+C:	Max (N_1 or N_2) x Max (S_1 or S_2) C

Fig. 13C

	1x16 real	2x16 real	1x16 cmpx	4x16 real	2x16 cmpx	1x32 real	2x32 real	1x32 cmpx	4x32 real	2x32 cmpx	1x40 real	2x40 real	1x40 real	4x40 real	2x40 cmpx
1x16 real	1 unit	2 unit	2 unit	4 unit	4 unit	2 unit	4 unit	4 unit							
2x16 real	2 unit	2 unit													
1x16 cmpx	2 unit		4 unit												
4x16 real	4 unit			4 unit											
2x16 cmpx	4 unit														
1x32 real	2 unit														
2x32 real	4 unit														
1x32 cmpx	4 unit														
4x32 real															
2x32 cmpx															
1x40 real															
2x40 real															
1x40 real															
4x40 real															
2x40 real															

FIG. 14

	1x16 real	2x16 real	1x16 cmpx	4x16 real	2x16 cmpx	1x32 real	2x32 real	1x32 cmpx	4x32 real	2x32 cmpx	1x40 real	2x40 real	1x40 real	4x40 real	2x40 cmpx
1x16 real	1 unit	2 unit		4 unit		1 unit	4 unit		4 unit		1 unit	2 unit		4 unit	
2x16 real	2 unit	2 unit				2 unit	2 unit					2 unit			
1x16 cmpx															
4x16 real	4 unit			4 unit		4 unit			4 unit					4 unit	
2x16 cmpx															
1x32 real	1 unit	2 unit		4 unit		1 unit	2 unit		4 unit		1 unit	2 unit		4 unit	
2x32 real	4 unit	2 unit				2 unit	2 unit					2 unit			
1x32 cmpx															
4x32 real	4 unit			4 unit		4 unit			4 unit		4 unit			4 unit	
2x32 cmpx															
1x40 real	1 unit					1 unit			4 unit		1 unit				
2x40 real	2 unit	2 unit				2 unit	2 unit					2 unit			
1x40 real															
4x40 real	4 unit		4 unit			4 unit			4 unit					4 unit	
2x40 real															

FIG. 15A

REPLACEMENT SHEET
Title: METHOD AND APPARATUS OF BUS STATE KEEPERS FOR BUSES IN AN INTEGRATED CIRCUIT
1st Named Inventor: Ruban Kanapathippillai
Application No.: 10/649,067 Docket No.: 42P14037D2
Sheet: 30/64

	1x16 real	2x16 real	1x16 cmpx	4x16 real	2x16 cmpx	1x32 real	2x32 real	1x32 cmpx	4x32 real	2x32 cmpx	1x40 real	2x40 real	1x40 real	4x40 real	2x40 cmpx
1x16 real	1 unit	2 unit	2 unit	4 unit	4 unit	1 unit	2 unit	2 unit	4 unit	4 unit	1 unit	2 unit	2 unit	4 unit	4 unit
2x16 real	2 unit	2 unit				2 unit	2 unit					2 unit			
1x16 cmpx	2 unit		2 unit					2 unit			2 unit		2 unit		
4x16 real	4 unit			1 unit		4 unit			4 unit					4 unit	
2x16 cmpx	4 unit				4 unit					4 unit					4 unit
1x32 real	1 unit	2 unit				1 unit	2 unit	2 unit	4 unit		1 unit	2 unit	2 unit	4 unit	
2x32 real	2 unit	2 unit				2 unit	2 unit					2 unit			
1x32 cmpx	2 unit		2 unit			2 unit		2 unit			2 unit		2 unit		
4x32 real	4 unit			4 unit		4 unit			4 unit		4 unit			4 unit	
2x32 cmpx	4 unit				4 unit					4 unit					4 unit
1x40 real	1 unit		2 unit			1 unit		2 unit	4 unit		1 unit	2 unit		4 unit	
2x40 real	2 unit	2 unit				2 unit	2 unit				2 unit	2 unit			
1x40 real	2 unit		2 unit			2 unit		2 unit					2 unit		
4x40 real	4 unit			4 unit		4 unit			4 unit		4 unit			4 unit	
2x40 real	4 unit				4 unit					4 unit					4 unit

FIG. 15B

REPLACEMENT SHEET
Title: METHOD AND APPARATUS OF BUS STATE KEEPERS FOR BUSES IN AN INTEGRATED CIRCUIT
1st Named Inventor: Ruban Kanapathippillai
Application No.: 10/649,067 Docket No.: 42P14037D2
Sheet: 31/64

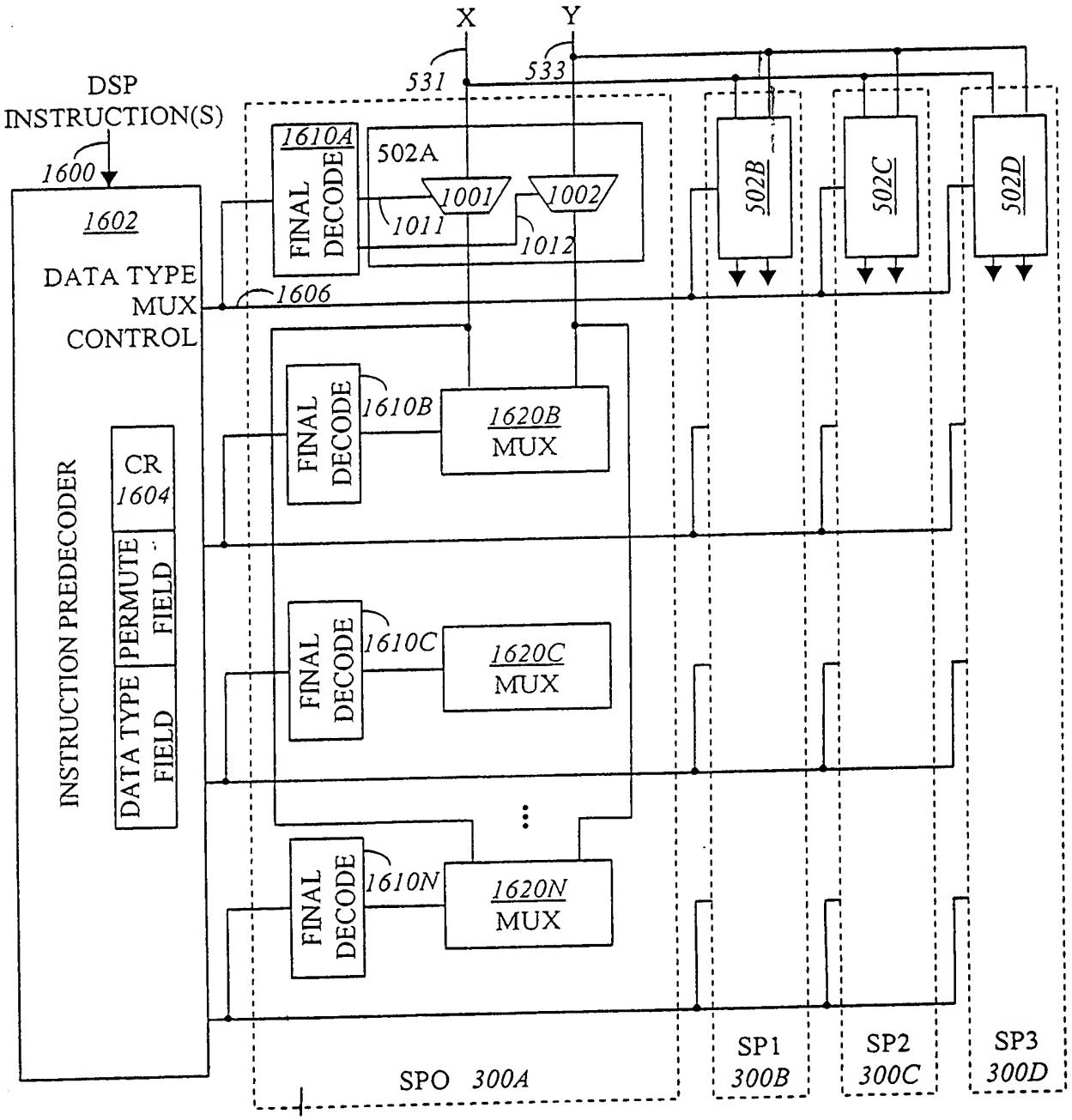

X
Y
531
533
DSP INSTRUCTION(S)
1600
1602
DATA TYPE MUX CONTROL
1606
INSTRUCTION PREDECODER
CR 1604
PERMUTE FIELD
DATA TYPE FIELD
1610A
FINAL DECODE
502A
1001
1002
1011
1012
502B
502C
502D
1610B
1620B MUX
1610C
1620C MUX
1610N
1620N MUX
SPO 300A
SP1 300B
SP2 300C
SP3 300D

FIG. 16

Data Type: N x S(R/C)

FIG. 17

(rX): Memory Address Registers

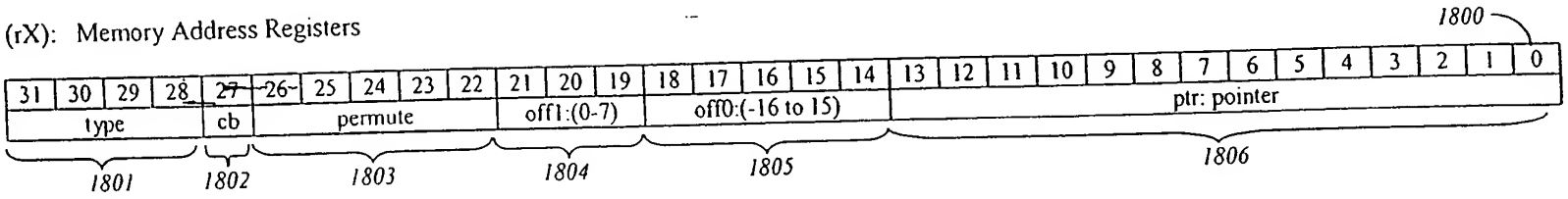


FIG. 18

Data Type *1801*

0000: 1x16 real
0001: 2x16 real
0010: 1x16 complex
0011: 4x16 real
0100: 1x32 real
0101: 2x32 real
0110: 1x32 complex
0111: 2x16 complex
1000: 4x32 real
1001: 2x32 complex
1010: 1x40 real
1011: 2x40 real
1100: 1x40 complex
1101: 4x40 real (only for local add unit operations)
1110: 2x40 complex (only for local add unit operations)
1111: Reserved

FIG. 19

REPLACEMENT SHEET
Title: METHOD AND APPARATUS OF BUS STATE KEEPERS FOR BUSES IN AN INTEGRATED CIRCUIT
1st Named Inventor: Ruban Kanapathippillai
Application No.: 10/649,067 Docket No.: 42P14037D2
Sheet: 33/64

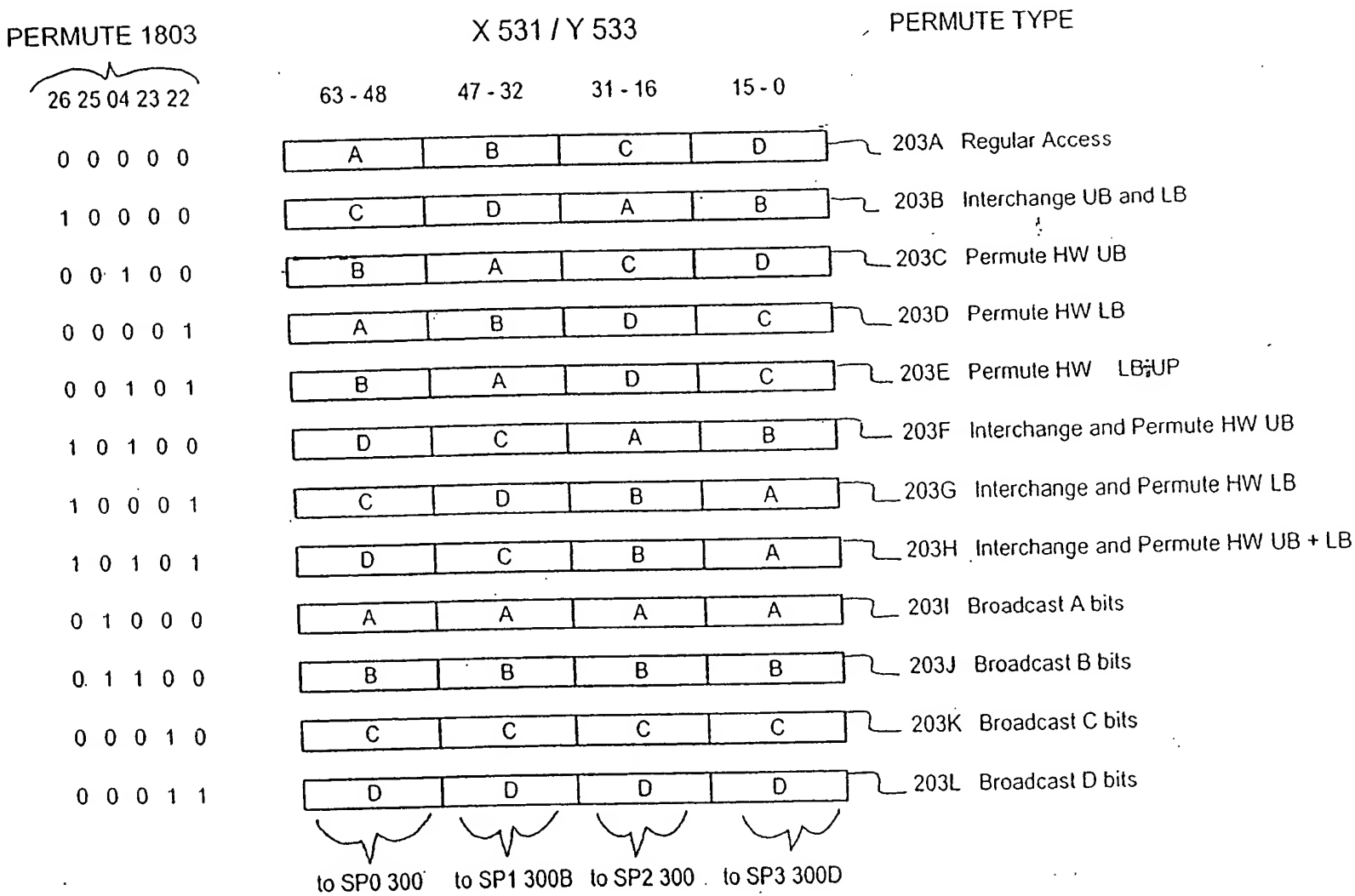

PERMUTE 1803
X 531 / Y 533
PERMUTE TYPE
26 25 04 23 22
63 - 48
47 - 32
31 - 16
15 - 0
0 0 0 0 0
A B C D
203A Regular Access
1 0 0 0 0
C D A B
203B Interchange UB and LB
0 0 1 0 0
B A C D
203C Permute HW UB
0 0 0 0 1
A B D C
203D Permute HW LB
0 0 1 0 1
B A D C
203E Permute HW LB:UP
1 0 1 0 0
D C A B
203F Interchange and Permute HW UB
1 0 0 0 1
C D B A
203G Interchange and Permute HW LB
1 0 1 0 1
D C B A
203H Interchange and Permute HW UB + LB
0 1 0 0 0
A A A A
203I Broadcast A bits
0 1 1 0 0
B B B B
203J Broadcast B bits
0 0 0 1 0
C C C C
203K Broadcast C bits
0 0 0 1 1
D D D D
203L Broadcast D bits
to SP0 300
to SP1 300B
to SP2 300
to SP3 300D

FIG. 20

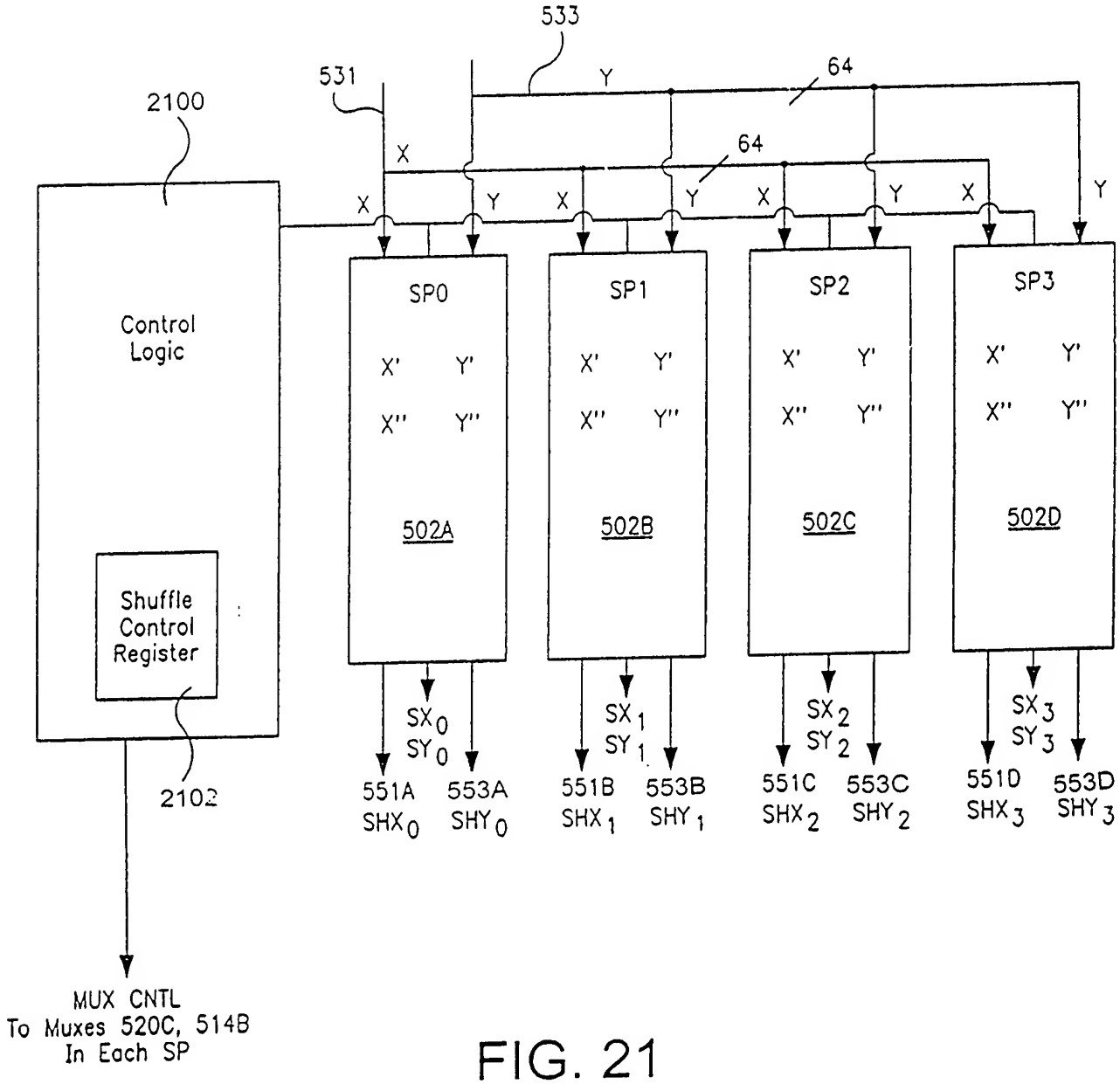

533
531
Y
64
2100
X
64
X
Y
X
Y
X
Y
X
Y
SP0
SP1
SP2
SP3
Control Logic
X' Y'
X'' Y''
502A
502B
502C
502D
Shuffle Control Register
SX0 SY0
SX1 SY1
SX2 SY2
SX3 SY3
2102
551A SHX0
553A SHY0
551B SHX1
553B SHY1
551C SHX2
553C SHY2
551D SHX3
553D SHY3
MUX CNTL
To Muxes 520C, 514B
In Each SP

FIG. 21

$X' = [\ SX_{10},\ SX_{11},\ SX_{12},\ SX_{13}\ ]$ e.g. $[\ X_0,\ X_1,\ X_2,\ X_3\ ]$

$X'' = [\ SX_{20},\ SX_{21},\ SX_{22},\ SX_{23}\ ]$ e.g. $[\ X_4,\ X_5,\ X_6,\ X_7\ ]$

Where SX_{ab} :S=Source; a=delay; b= SP unit number(e.g. SP3,SP2 SP1,SP0; or termed U3,U2,U1,U0)

$Y' = [\ SY_{10},\ SY_{11},\ SY_{12},\ SY_{13}\ ]$

$Y'' = [\ SY_{20},\ SY_{21},\ SY_{22},\ SY_{23}\ ]$

Where SY_{ab} :S=Source; a=delay; b= SP unit number(e.g. SP3,SP2,SP1, SP0; or termed U3,U1,U0)

FIG. 22A

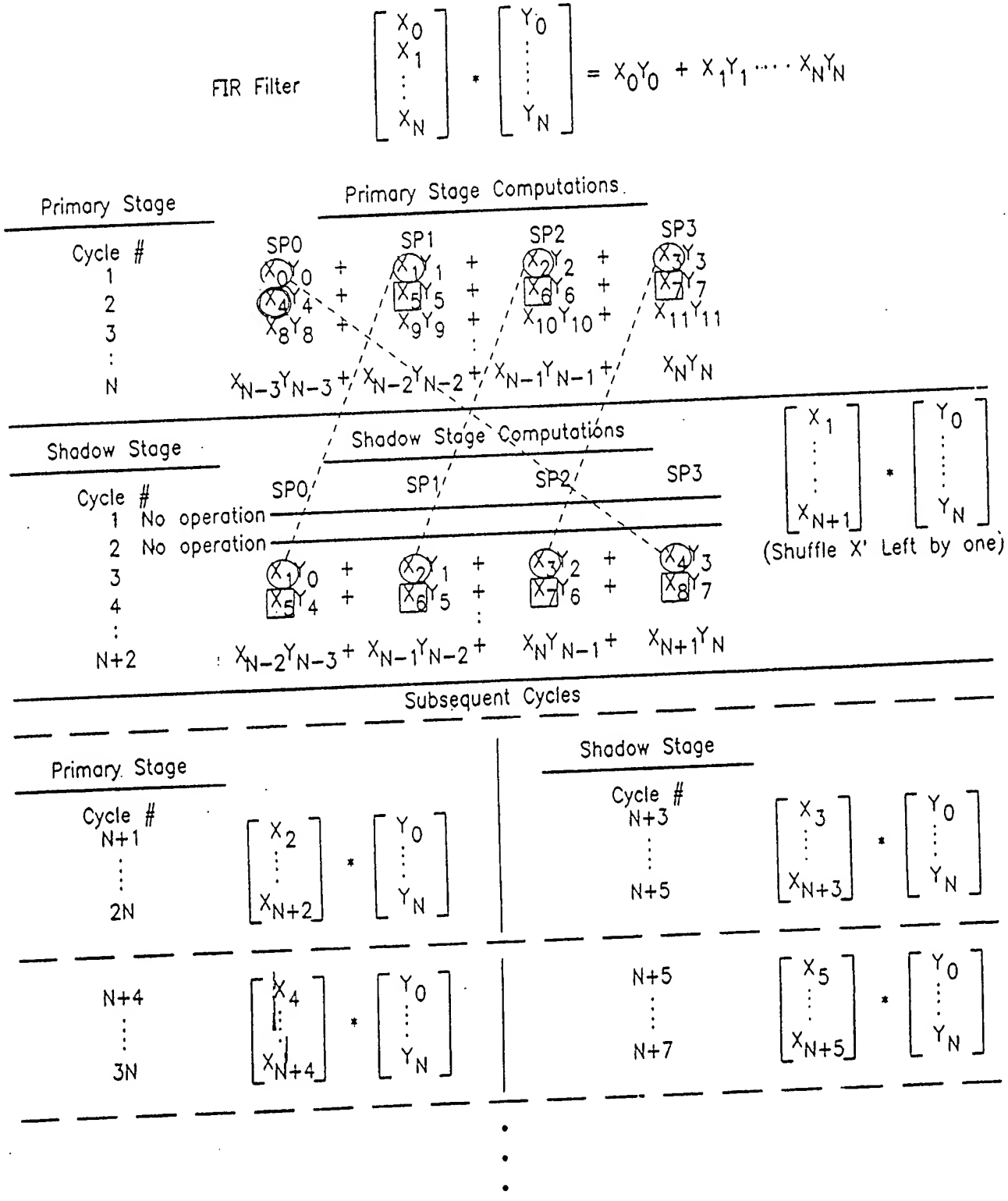

FIR Filter
Primary Stage
Primary Stage Computations
Cycle #
SP0
SP1
SP2
SP3
Shadow Stage
Shadow Stage Computations
No operation
No operation
(Shuffle X' Left by one)
Subsequent Cycles
Primary Stage
Shadow Stage

FIG. 22B

shuffle Shuffle Control Register

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
u3		u2		u1		u0		u3		u2		u1		u0		u3		u2		u1		u0		u3		u2		u1		u0	
SY2S								SY1S								SX2S								SX1S							

Units are connected to their nearest neighbors for shuffling the sources using the following bit diagram:

00 Unit N+1, SX1=X' (right)
01 Unit N+1, SX2=X" (right)
10 Unit N−1, SX1=X' (left)
11 Unit N−1, SX2=X" (left)

For example to shift the sources to the left by one:

3	2	1	0	From
2	1	0	3	Into

The bits should be 10101010 ($AA)

FIG. 22C

REPLACEMENT SHEET
Title: METHOD AND APPARATUS OF BUS STATE KEEPERS FOR BUSES IN AN INTEGRATED CIRCUIT
1st Named Inventor: Ruban Kanapathippillai
Application No.: 10/649,067 Docket No.: 42P14037D2
Sheet: 38/64

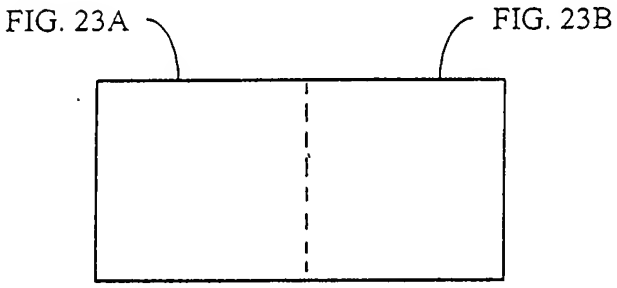


FIG. 23

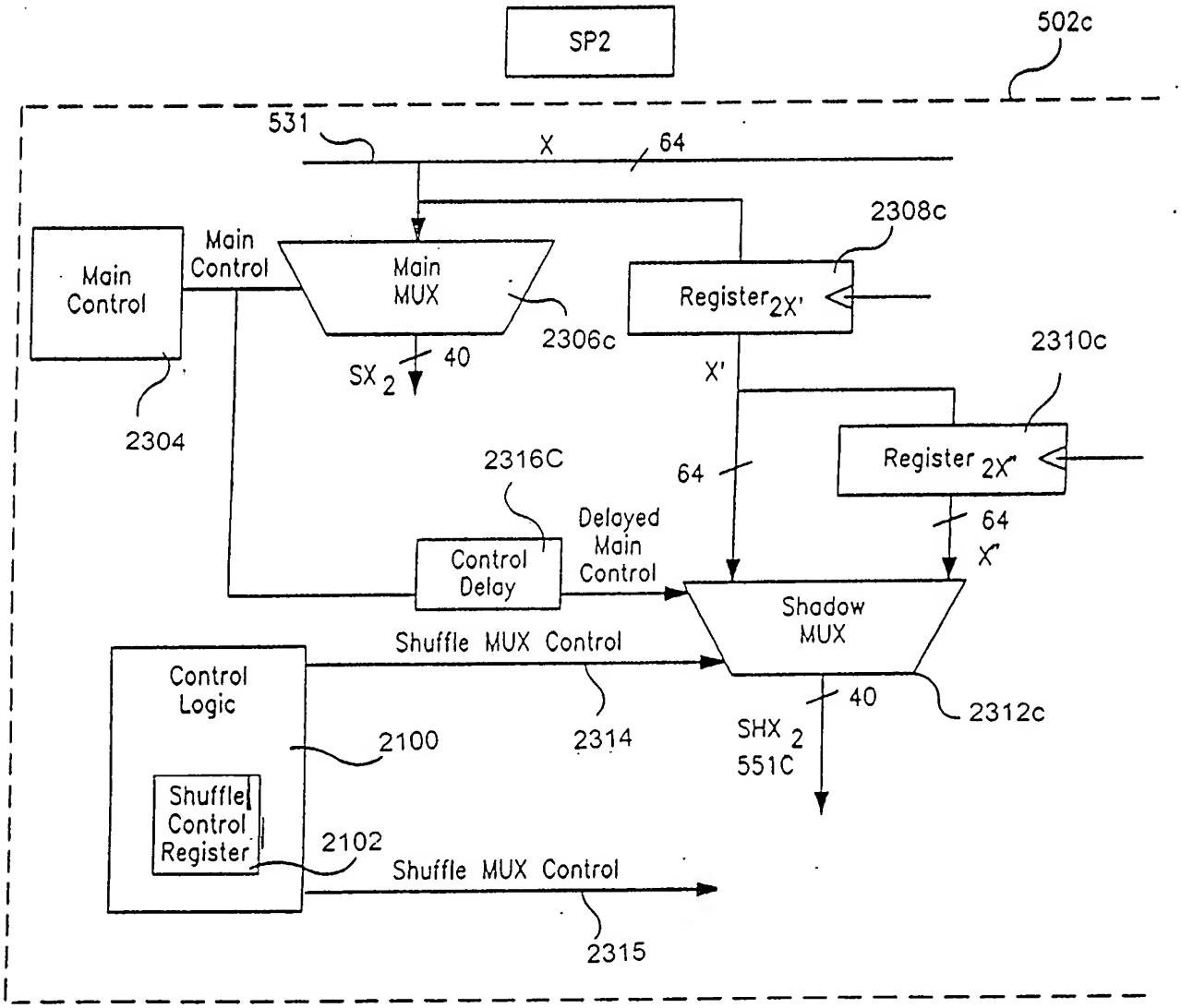


FIG. 23A

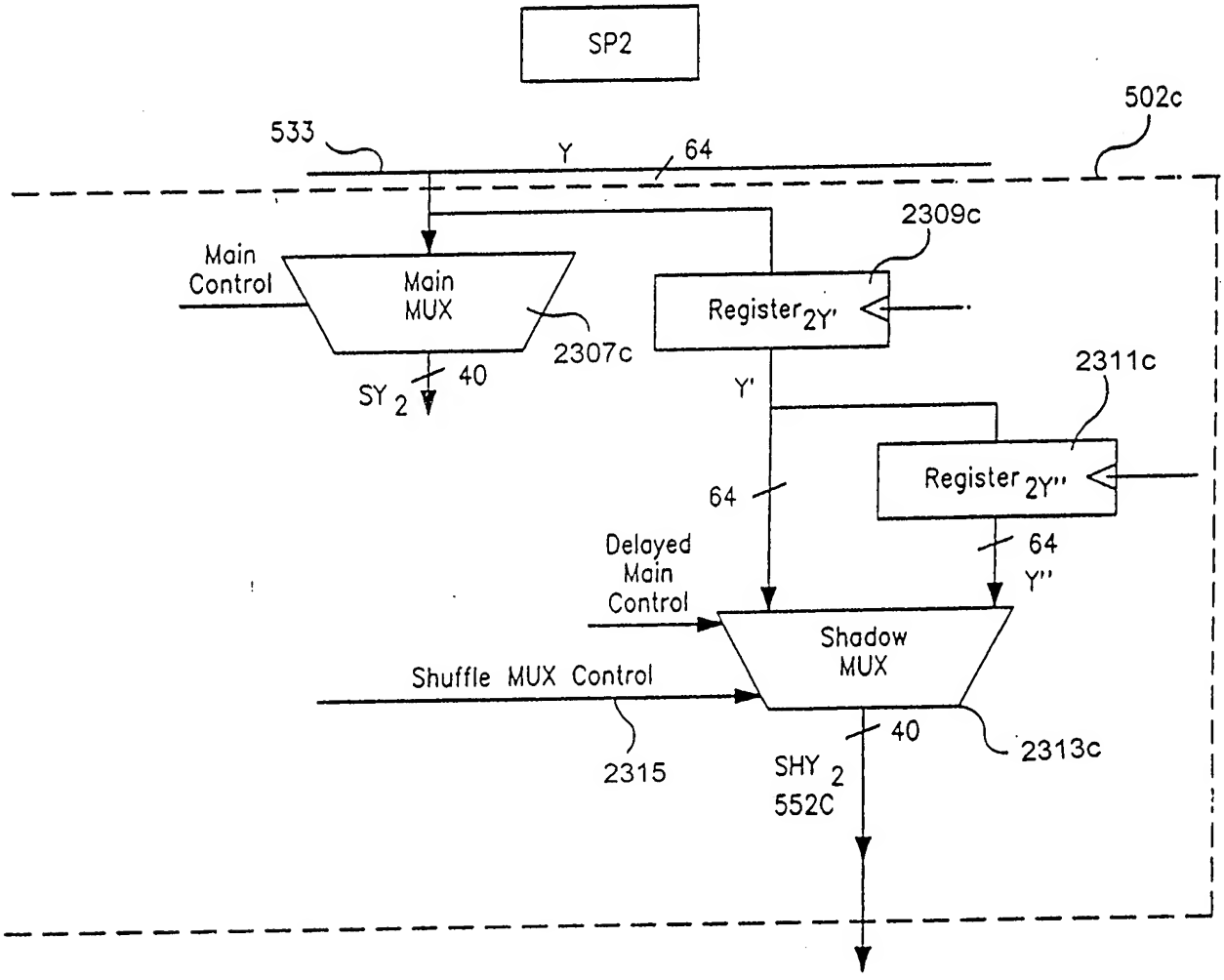

SP2
502c
533
Y
64
2309c
Main Control
Main MUX
2307c
Register 2Y'
SY 2
40
Y'
2311c
64
Register 2Y''
Delayed Main Control
64
Y''
Shadow MUX
Shuffle MUX Control
2315
40
2313c
SHY 2
552C

FIG. 23B

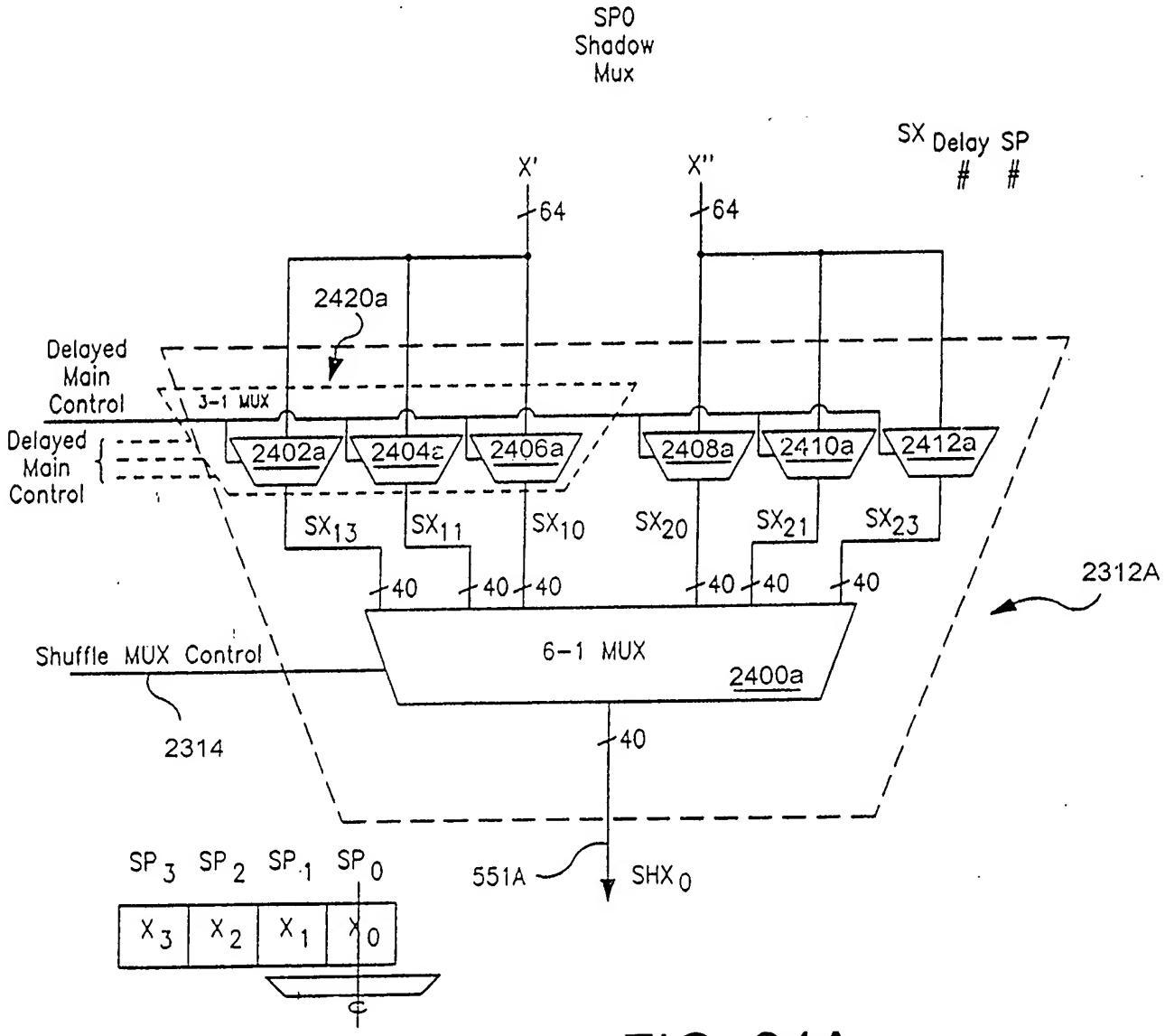


FIG. 24A

$X_0 = SX_{10}, SX_{20}$
$X_1 = SX_{11}, SX_{21}$
$X_3 = SX_{13}, SX_{23}$

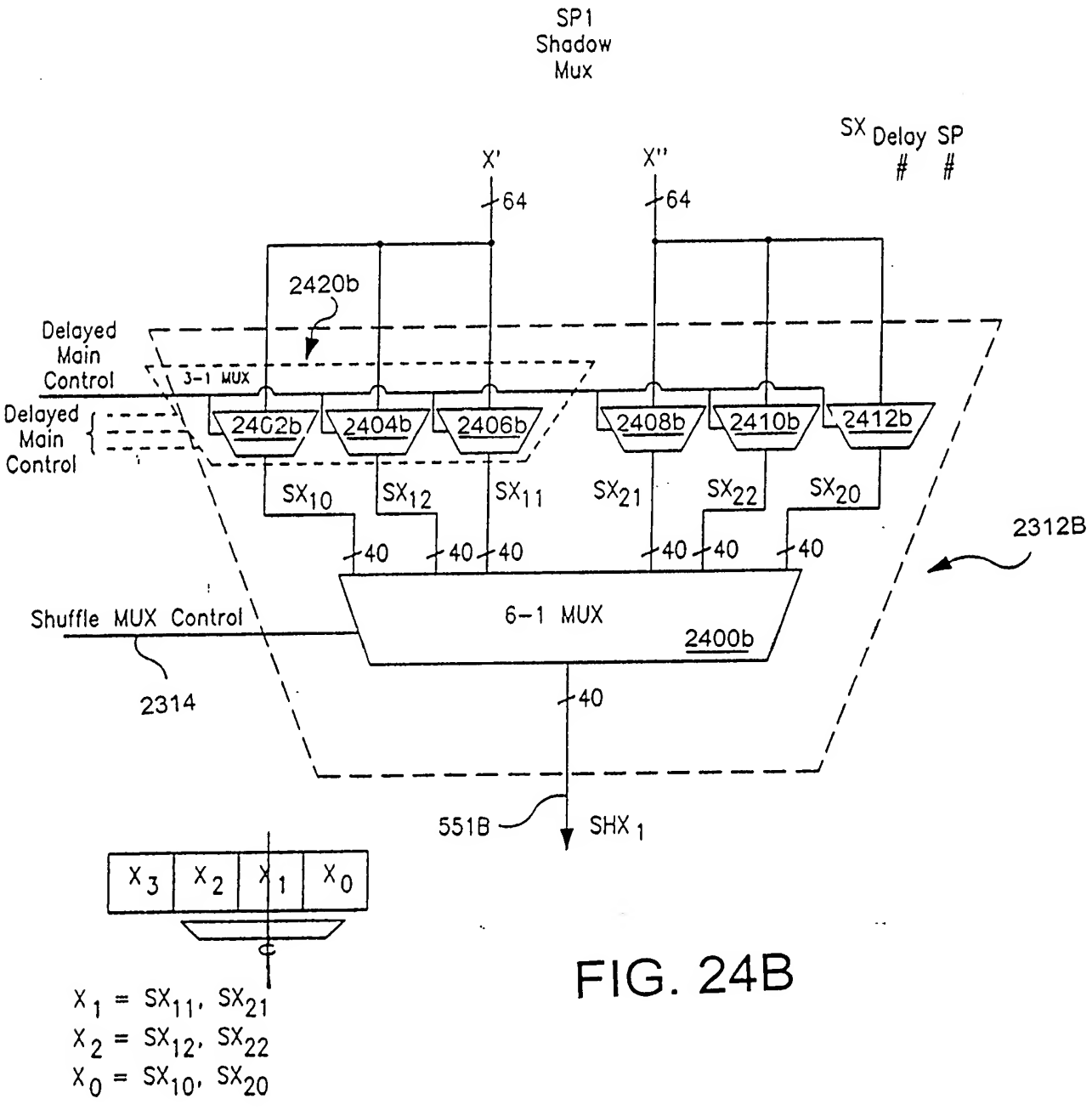
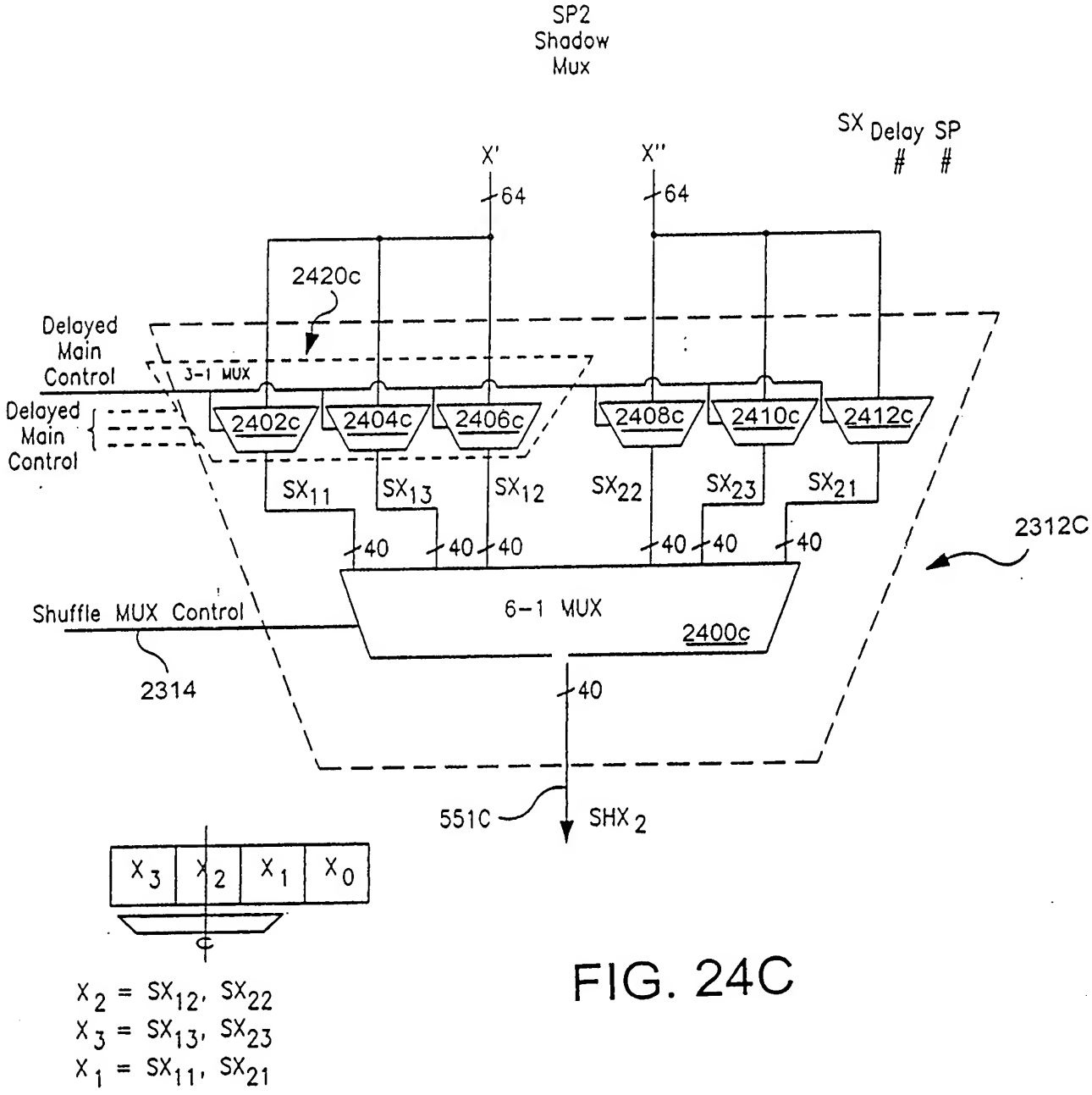

SP1
Shadow
Mux
SX Delay SP
# #
X'
X''
64
64
2420b
Delayed
Main
Control
3-1 MUX
Delayed
Main
Control
2402b
2404b
2406b
2408b
2410b
2412b
SX10
SX12
SX11
SX21
SX22
SX20
40
40
40
40
40
40
2312B
Shuffle MUX Control
6-1 MUX
2400b
2314
40
551B
SHX1
X3
X2
X1
X0
X1 = SX11, SX21
X2 = SX12, SX22
X0 = SX10, SX20

FIG. 24B



$X_2 = SX_{12}, SX_{22}$
$X_3 = SX_{13}, SX_{23}$
$X_1 = SX_{11}, SX_{21}$

FIG. 24C

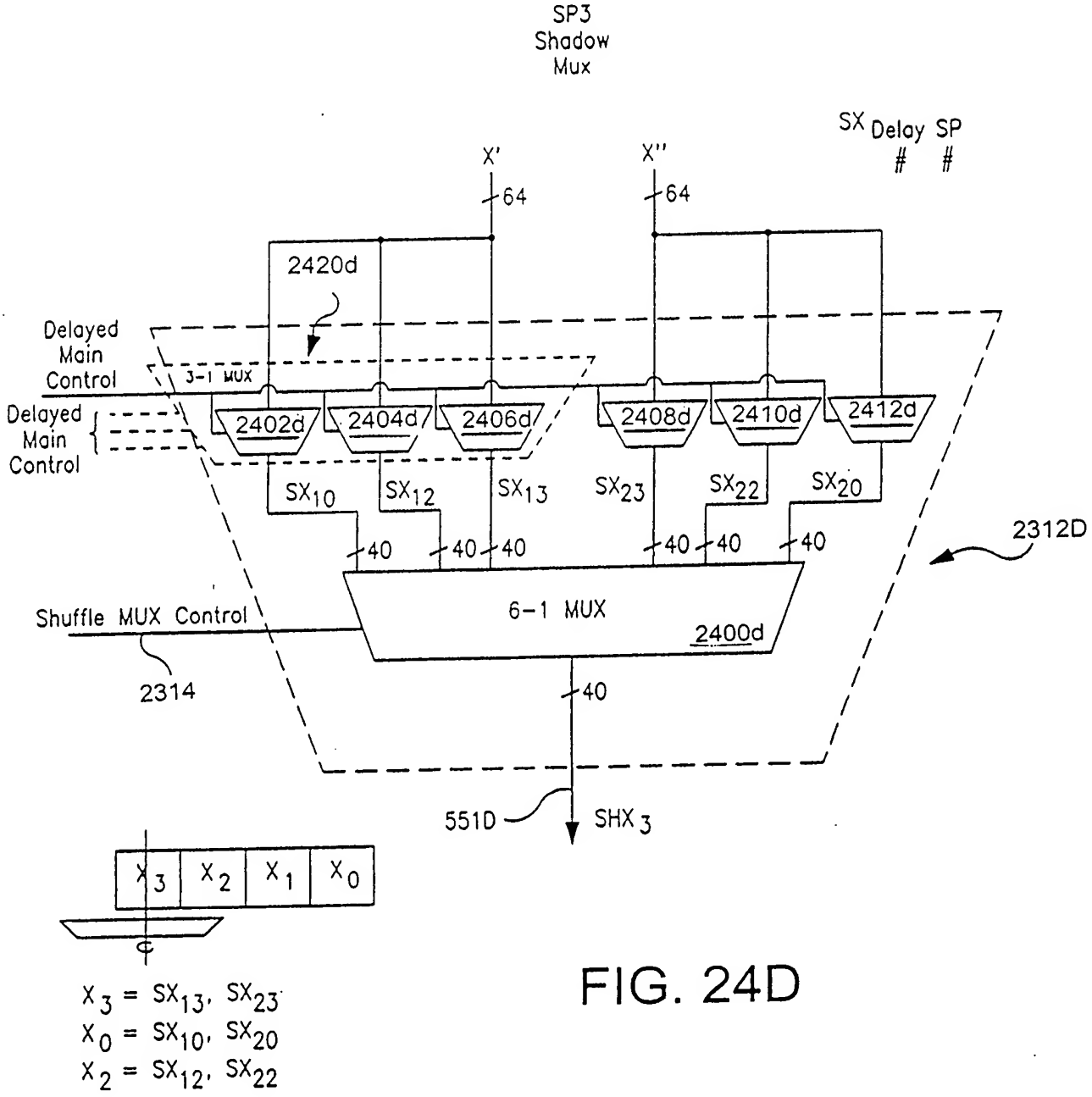

SP3
Shadow
Mux
SX Delay SP
# #
X'
X''
64
64
2420d
Delayed
Main
Control
3-1 MUX
Delayed
Main
Control
2402d
2404d
2406d
2408d
2410d
2412d
SX10
SX12
SX13
SX23
SX22
SX20
40
40
40
40
40
40
2312D
Shuffle MUX Control
6-1 MUX
2400d
2314
40
551D
SHX3
X3
X2
X1
X0
X3 = SX13, SX23
X0 = SX10, SX20
X2 = SX12, SX22

FIG. 24D

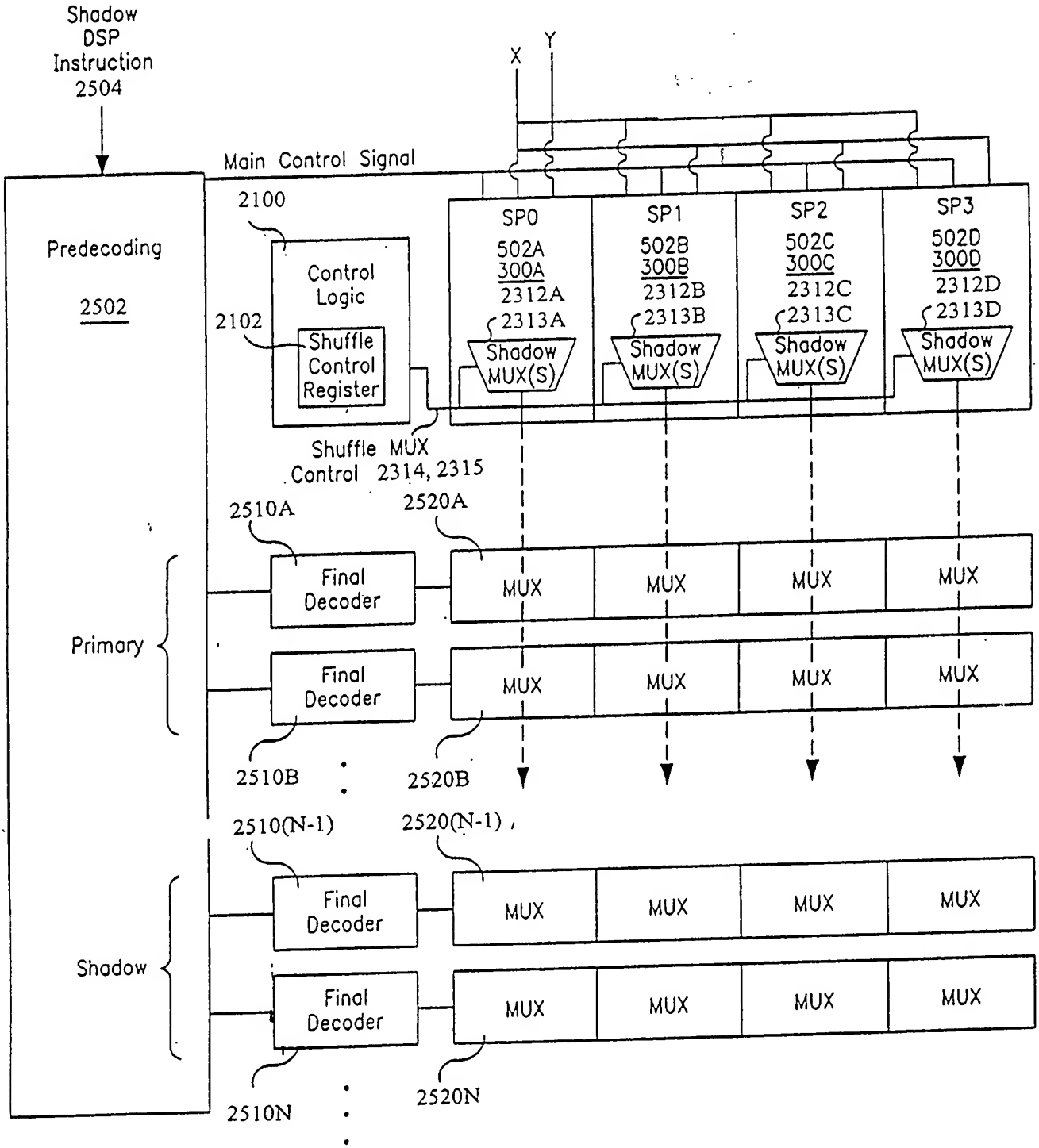

Shadow DSP Instruction 2504
Predecoding 2502
Main Control Signal
X
Y
2100
Control Logic
2102
Shuffle Control Register
Shuffle MUX Control 2314, 2315
SP0
502A
300A
2312A
2313A
Shadow MUX(S)
SP1
502B
300B
2312B
2313B
Shadow MUX(S)
SP2
502C
300C
2312C
2313C
Shadow MUX(S)
SP3
502D
300D
2312D
2313D
Shadow MUX(S)
2510A
2520A
Final Decoder
MUX
Primary
2510B
2520B
2510(N-1)
2520(N-1)
Shadow
2510N
2520N

FIG. 25

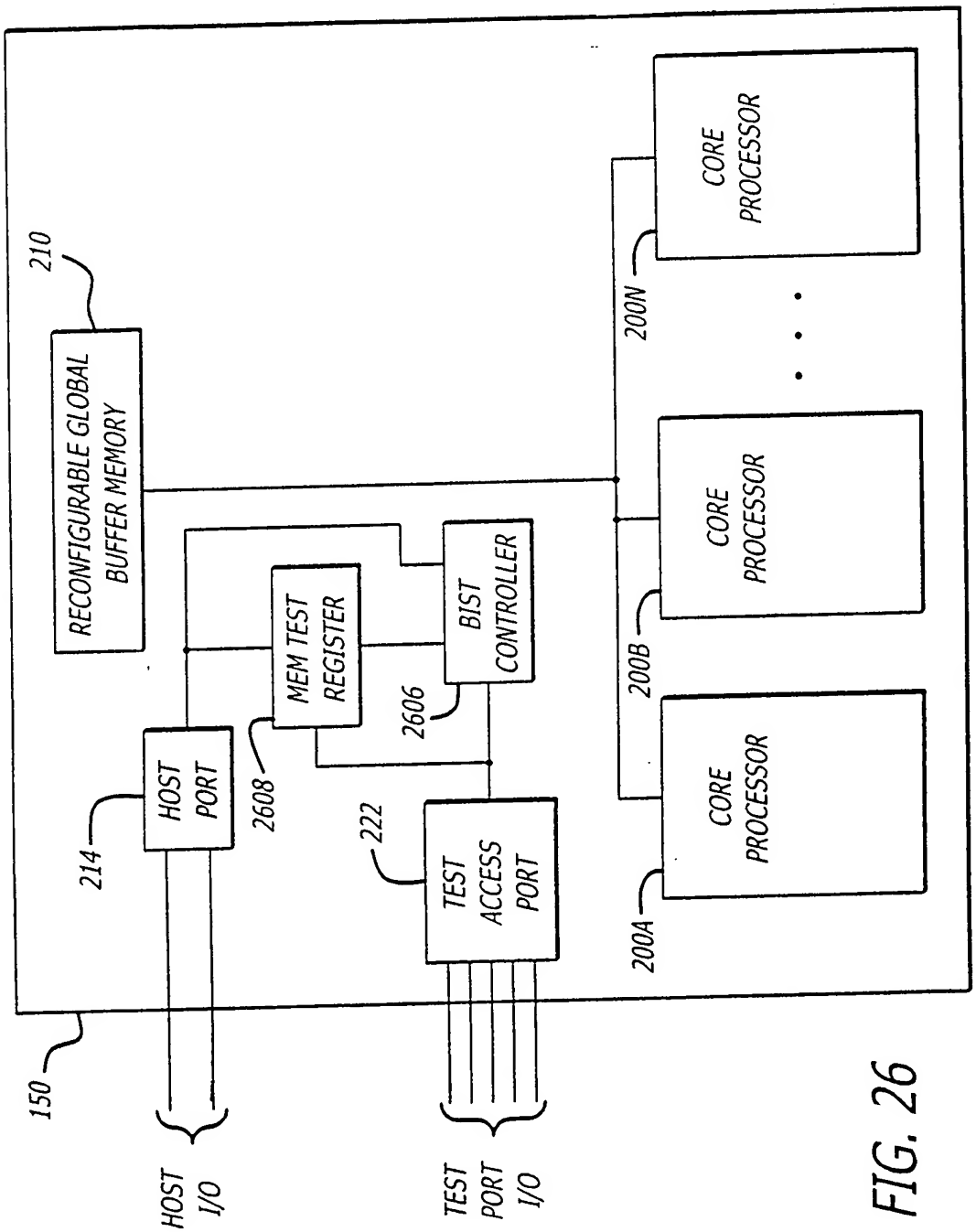

210
RECONFIGURABLE GLOBAL BUFFER MEMORY
MEM TEST REGISTER
BIST CONTROLLER
2606
2608
HOST PORT
214
222
TEST ACCESS PORT
150
HOST I/O
TEST PORT I/O
CORE PROCESSOR
200N
CORE PROCESSOR
200B
CORE PROCESSOR
200A

FIG. 26

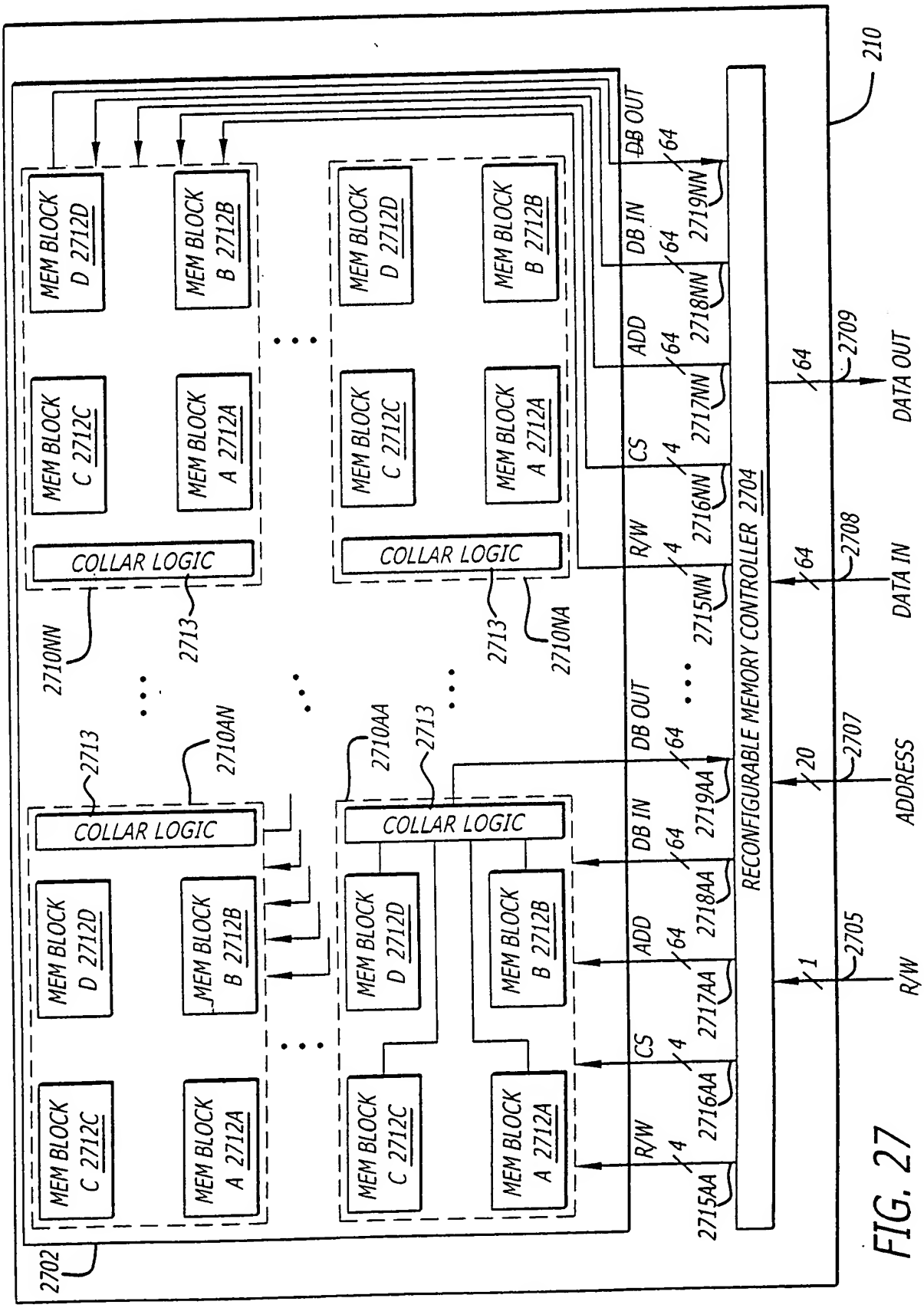

210
2702
MEM BLOCK D 2712D
MEM BLOCK B 2712B
MEM BLOCK C 2712C
MEM BLOCK A 2712A
COLLAR LOGIC
2710NN
2710AN
2710AA
2710NA
2713
DB OUT
DB IN
ADD
CS
R/W
2719NN
2718NN
2717NN
2716NN
2715NN
2719AA
2718AA
2717AA
2716AA
2715AA
64
4
20
1
RECONFIGURABLE MEMORY CONTROLLER 2704
2709
DATA OUT
2708
DATA IN
2707
ADDRESS
2705
R/W

FIG. 27

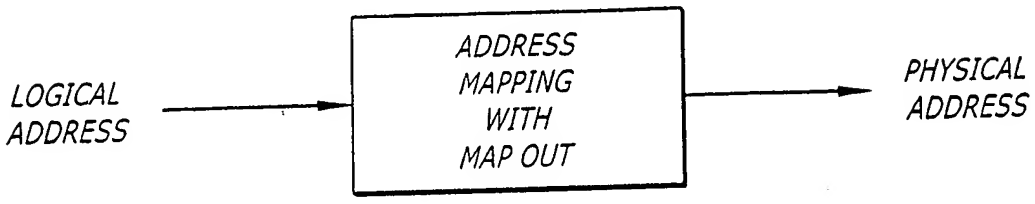


FIG. 28

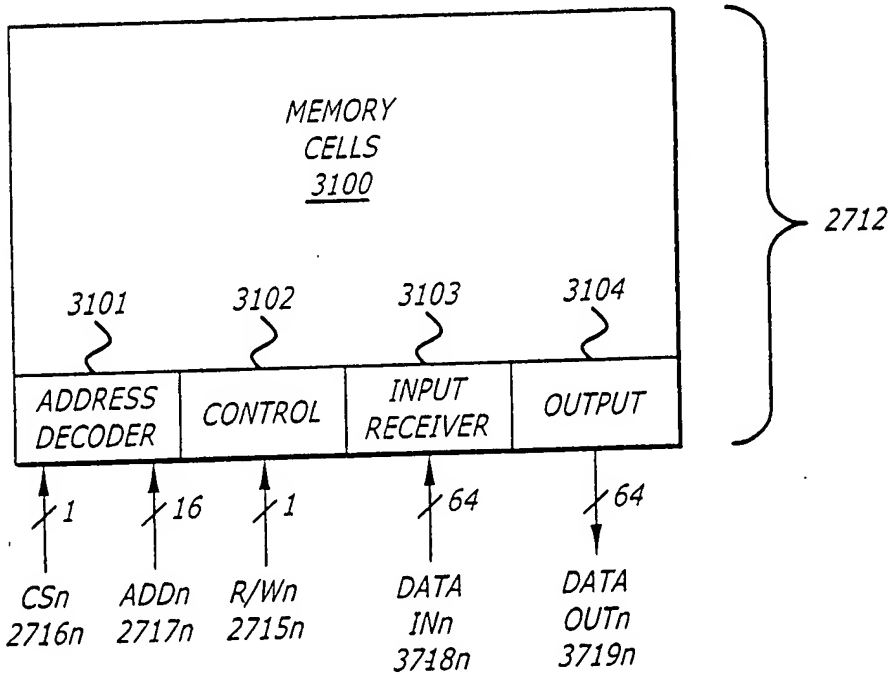


FIG. 31

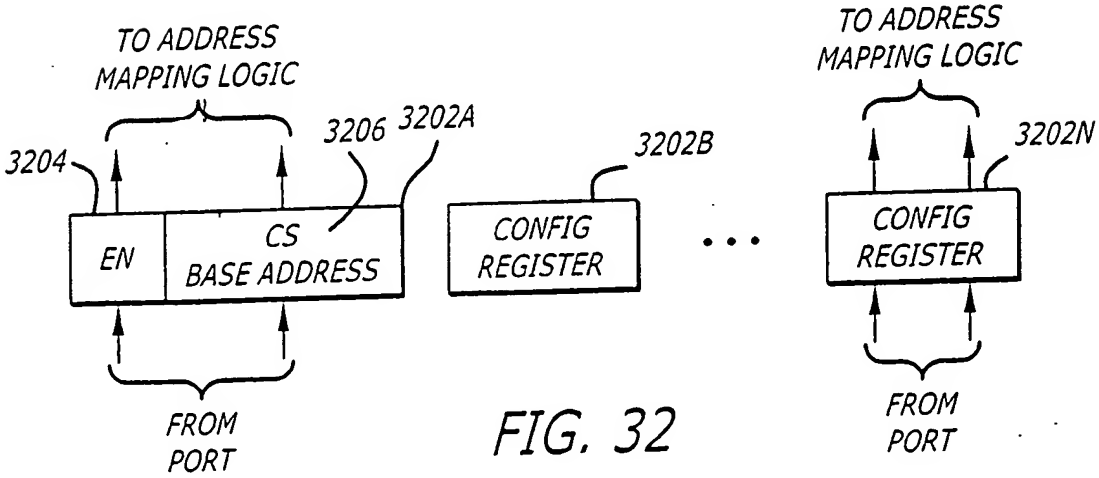
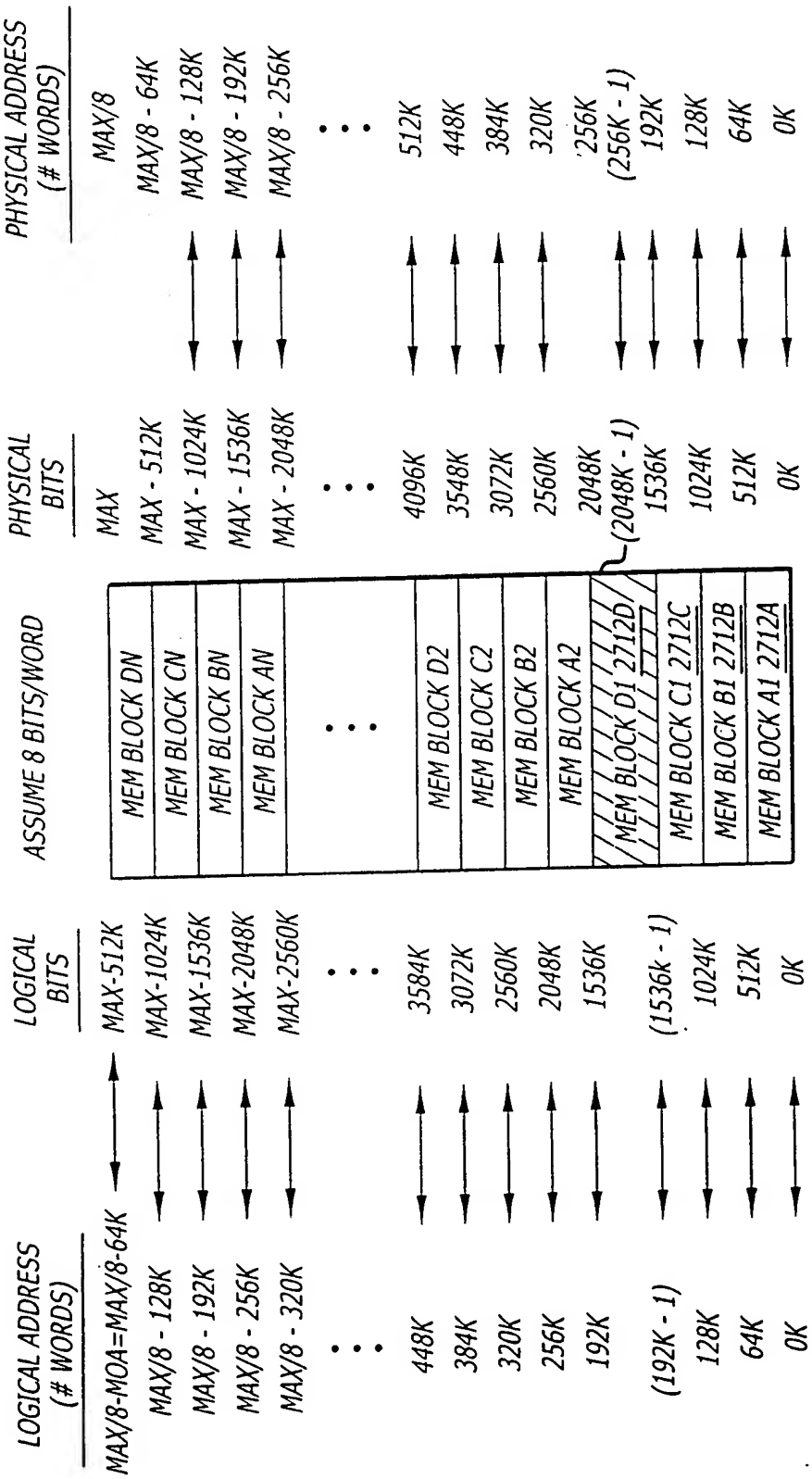


FIG. 32

FIG. 29


LOGICAL ADDRESS (# WORDS)
MAX/8-MOA=MAX/8-64K
MAX/8 - 128K
MAX/8 - 192K
MAX/8 - 256K
MAX/8 - 320K
448K
384K
320K
256K
192K
(192K - 1)
128K
64K
0K
LOGICAL BITS
MAX-512K
MAX-1024K
MAX-1536K
MAX-2048K
MAX-2560K
3584K
3072K
2560K
2048K
1536K
(1536k - 1)
1024K
512K
0K
ASSUME 8 BITS/WORD
MEM BLOCK DN
MEM BLOCK CN
MEM BLOCK BN
MEM BLOCK AN
MEM BLOCK D2
MEM BLOCK C2
MEM BLOCK B2
MEM BLOCK A2
MEM BLOCK D1 2712D
MEM BLOCK C1 2712C
MEM BLOCK B1 2712B
MEM BLOCK A1 2712A
PHYSICAL BITS
MAX
MAX - 512K
MAX - 1024K
MAX - 1536K
MAX - 2048K
4096K
3548K
3072K
2560K
2048K
(2048K - 1)
1536K
1024K
512K
0K
PHYSICAL ADDRESS (# WORDS)
MAX/8
MAX/8 - 64K
MAX/8 - 128K
MAX/8 - 192K
MAX/8 - 256K
512K
448K
384K
320K
256K
(256K - 1)
192K
128K
64K
0K

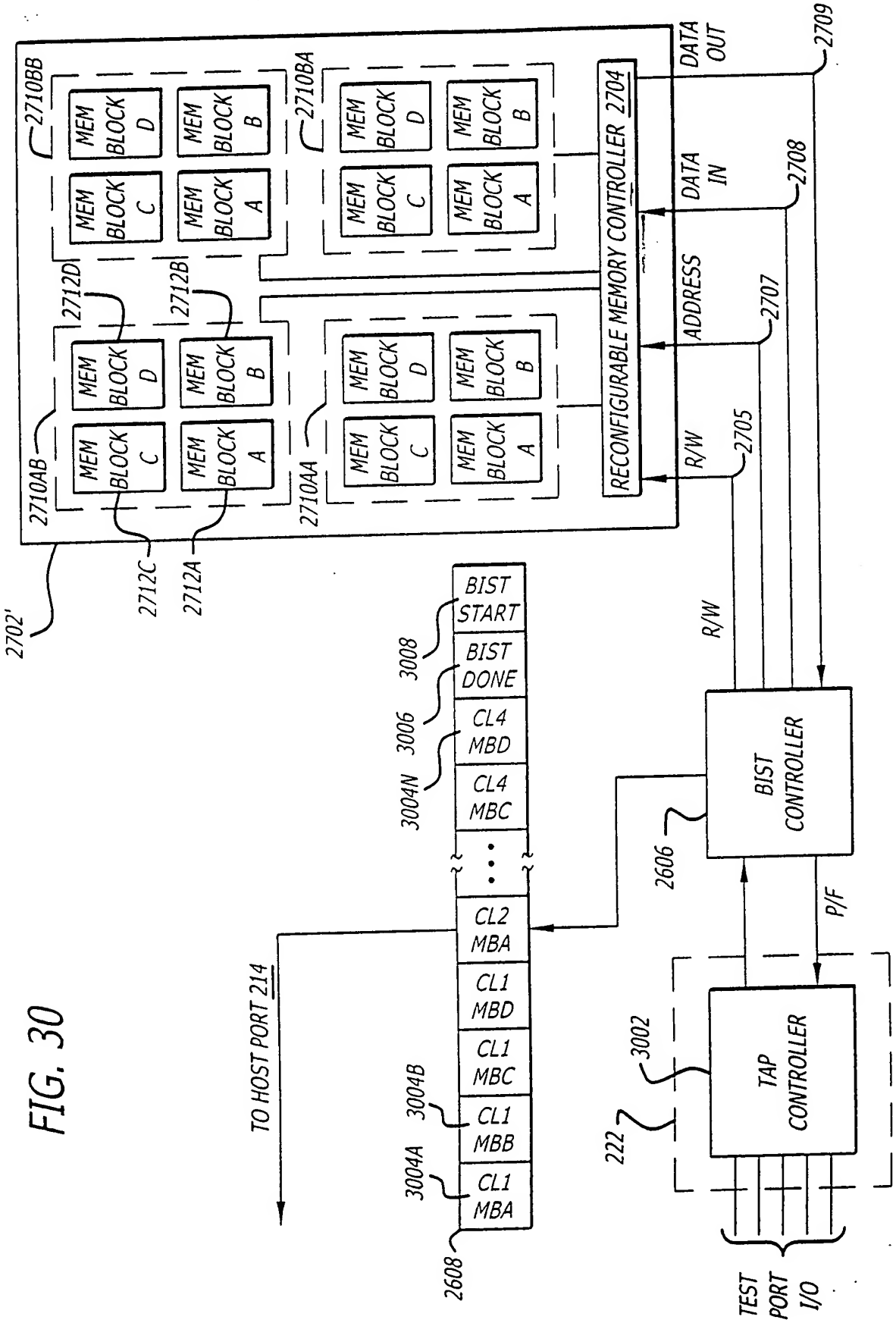

2710BB
2710BA
2710AB
2710AA
MEM BLOCK D
MEM BLOCK C
MEM BLOCK B
MEM BLOCK A
2712D
2712B
2712C
2712A
2702'
RECONFIGURABLE MEMORY CONTROLLER 2704
DATA OUT
2709
DATA IN
2708
ADDRESS
2707
R/W
2705
BIST START
BIST DONE
CL4 MBD
CL4 MBC
CL2 MBA
CL1 MBD
CL1 MBC
CL1 MBB
CL1 MBA
3008
3006
3004N
3004B
3004A
2608
TO HOST PORT 214
BIST CONTROLLER
2606
P/F
TAP CONTROLLER
3002
222
TEST PORT I/O

FIG. 30

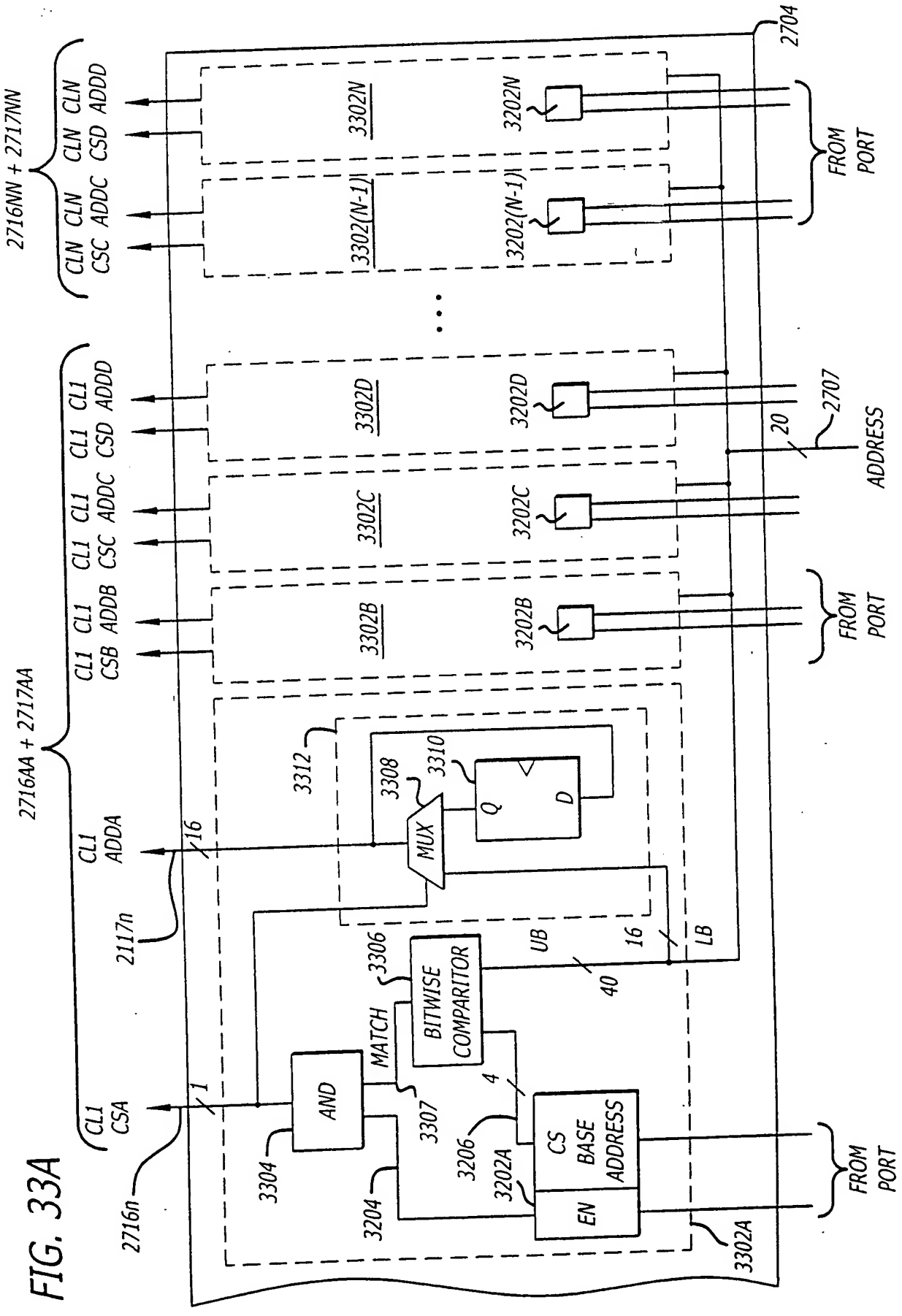

2716NN + 2717NN
CLN CSC
CLN ADDC
CLN CSD
CLN ADDD
3302N
3202N
3302(N-1)
3202(N-1)
2704
FROM PORT
2716AA + 2717AA
CL1 CSB
CL1 ADDB
CL1 CSC
CL1 ADDC
CL1 CSD
CL1 ADDD
3302D
3202D
3302C
3202C
3302B
3202B
20
2707
ADDRESS
CL1 ADDA
16
2117n
3312
3308
3310
MUX
Q
D
3306
BITWISE COMPARITOR
UB
LB
16
40
CL1 CSA
1
2716n
AND
MATCH
3307
4
3304
3204
3206
3202A
CS
BASE ADDRESS
EN
3302A

FIG. 33A

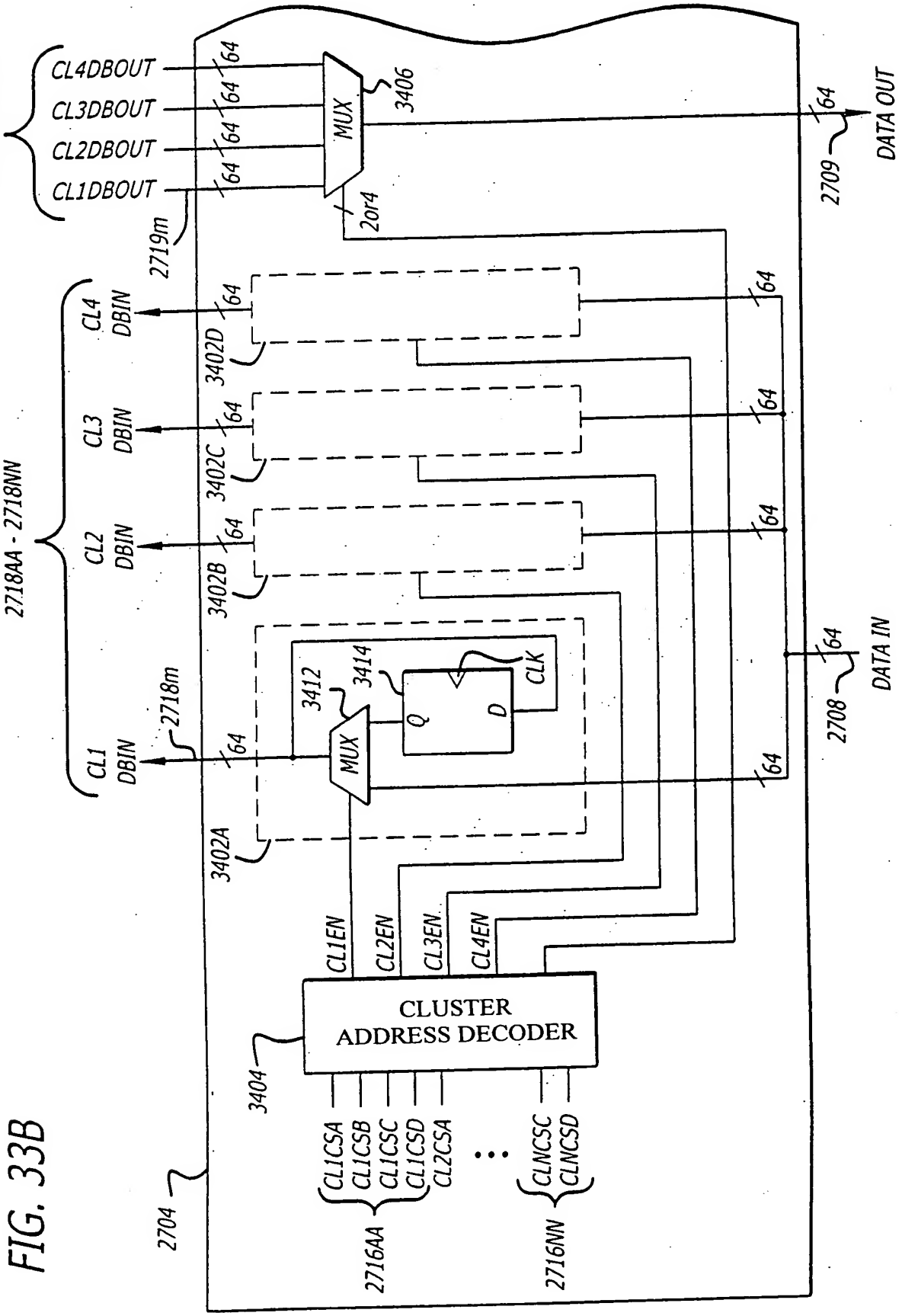

CL4DBOUT
CL3DBOUT
CL2DBOUT
CL1DBOUT
2719AA - 2719NN
2719m
64
MUX
3406
2or4
DATA OUT
2709
CL4 DBIN
CL3 DBIN
CL2 DBIN
CL1 DBIN
2718AA - 2718NN
2718m
3402D
3402C
3402B
3402A
3412
3414
Q
D
CLK
DATA IN
2708
CL1EN
CL2EN
CL3EN
CL4EN
CLUSTER ADDRESS DECODER
3404
CL1CSA
CL1CSB
CL1CSC
CL1CSD
CL2CSA
CLNCSC
CLNCSD
2716AA
2716NN
2704

FIG. 33B

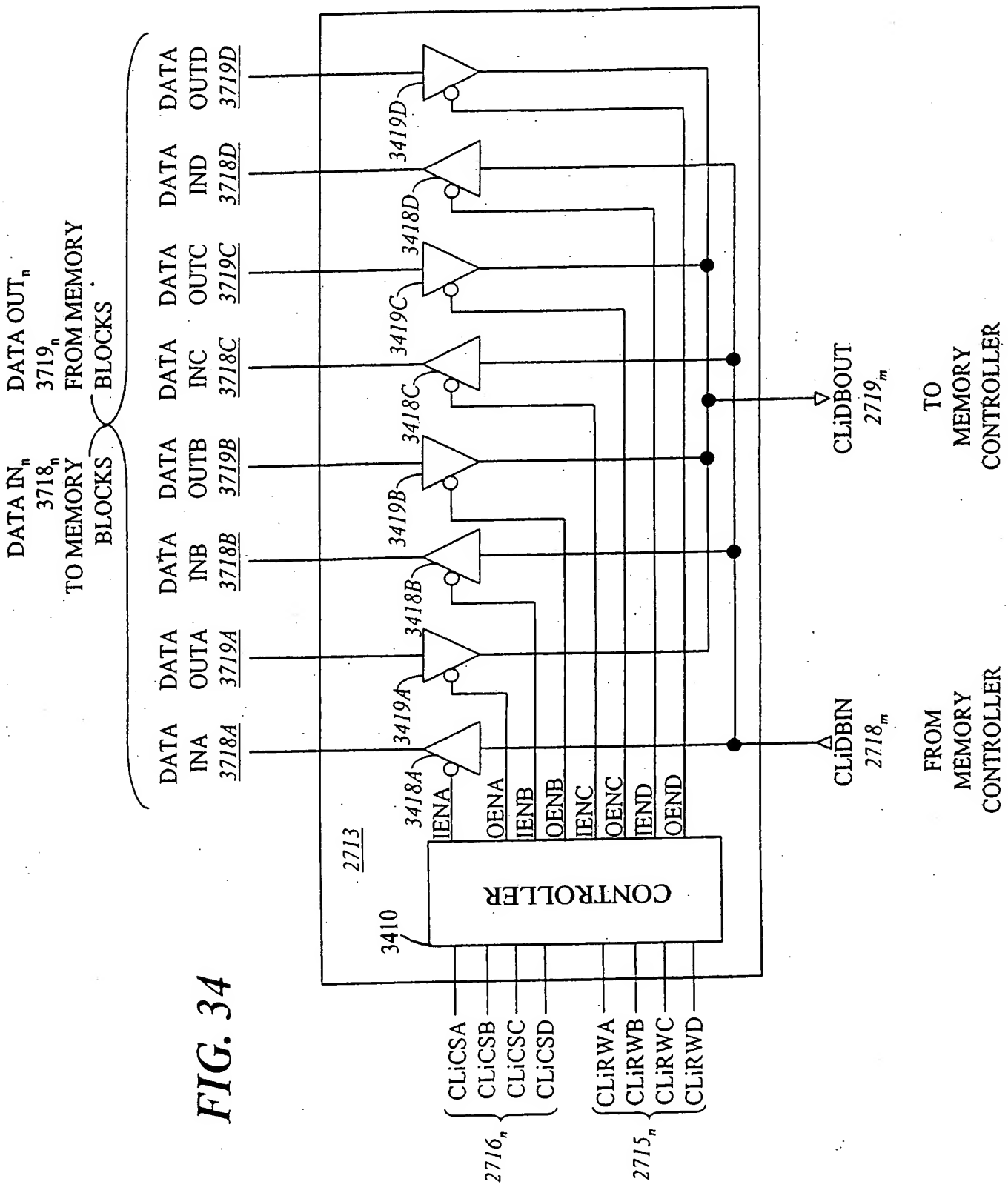

DATA IN$_n$ 3718$_n$ TO MEMORY BLOCKS
DATA OUT$_n$ 3719$_n$ FROM MEMORY BLOCKS
DATA INA 3718A
DATA OUTA 3719A
DATA INB 3718B
DATA OUTB 3719B
DATA INC 3718C
DATA OUTC 3719C
DATA IND 3718D
DATA OUTD 3719D
3418A
3418B
3418C
3418D
3419A
3419B
3419C
3419D
2713
3410
CONTROLLER
IENA
OENA
IENB
OENB
IENC
OENC
IEND
OEND
CLiCSA
CLiCSB
CLiCSC
CLiCSD
CLiRWA
CLiRWB
CLiRWC
CLiRWD
2716$_n$
2715$_n$
CLiDBIN 2718$_m$
CLiDBOUT 2719$_m$
FROM MEMORY CONTROLLER
TO MEMORY CONTROLLER

FIG. 34

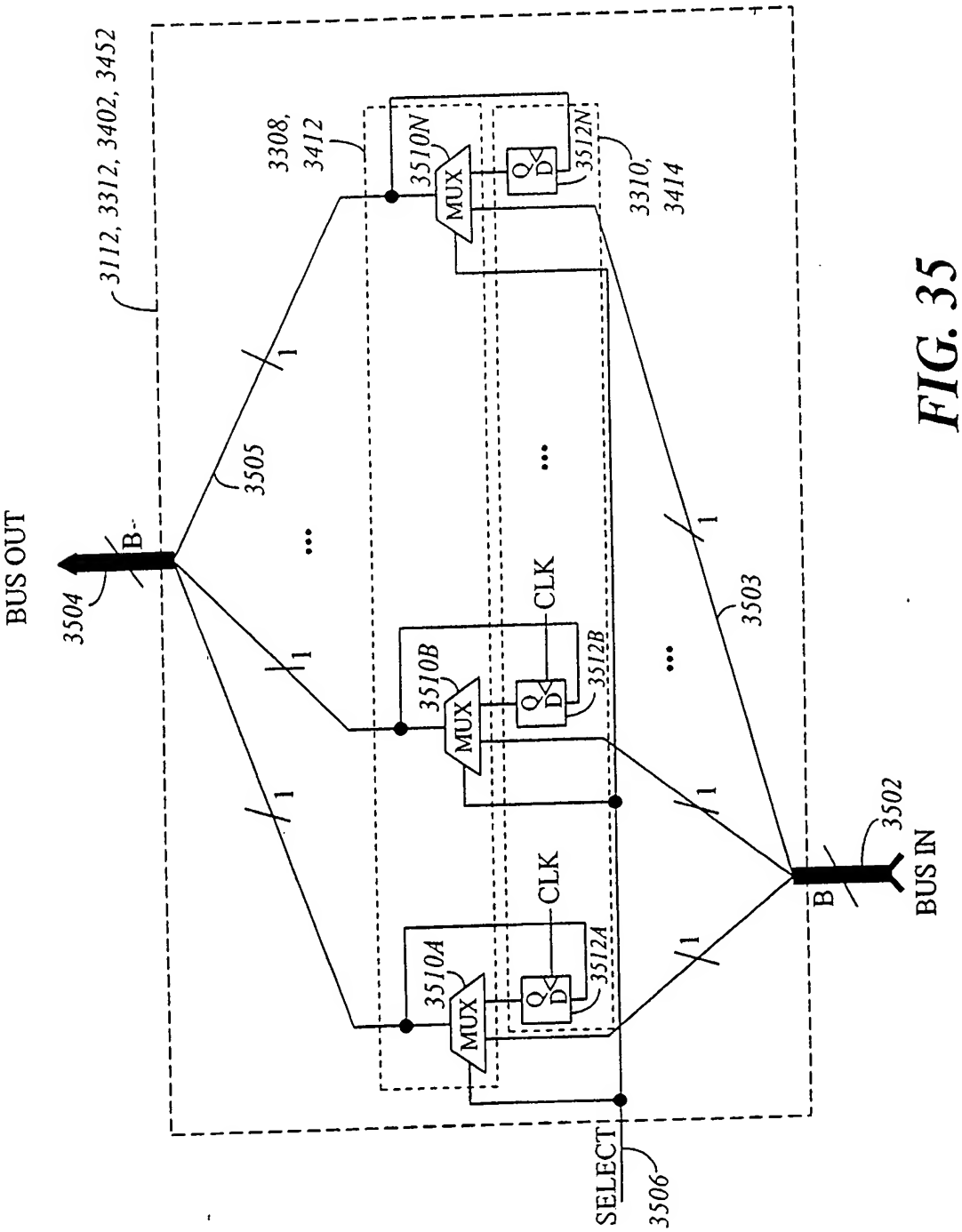

3112, 3312, 3402, 3452
3308, 3412
3310, 3414
3510N
3512N
MUX
Q
D
CLK
3510B
3512B
3510A
3512A
3505
3503
3504
BUS OUT
B
3502
BUS IN
SELECT
3506
1

FIG. 35

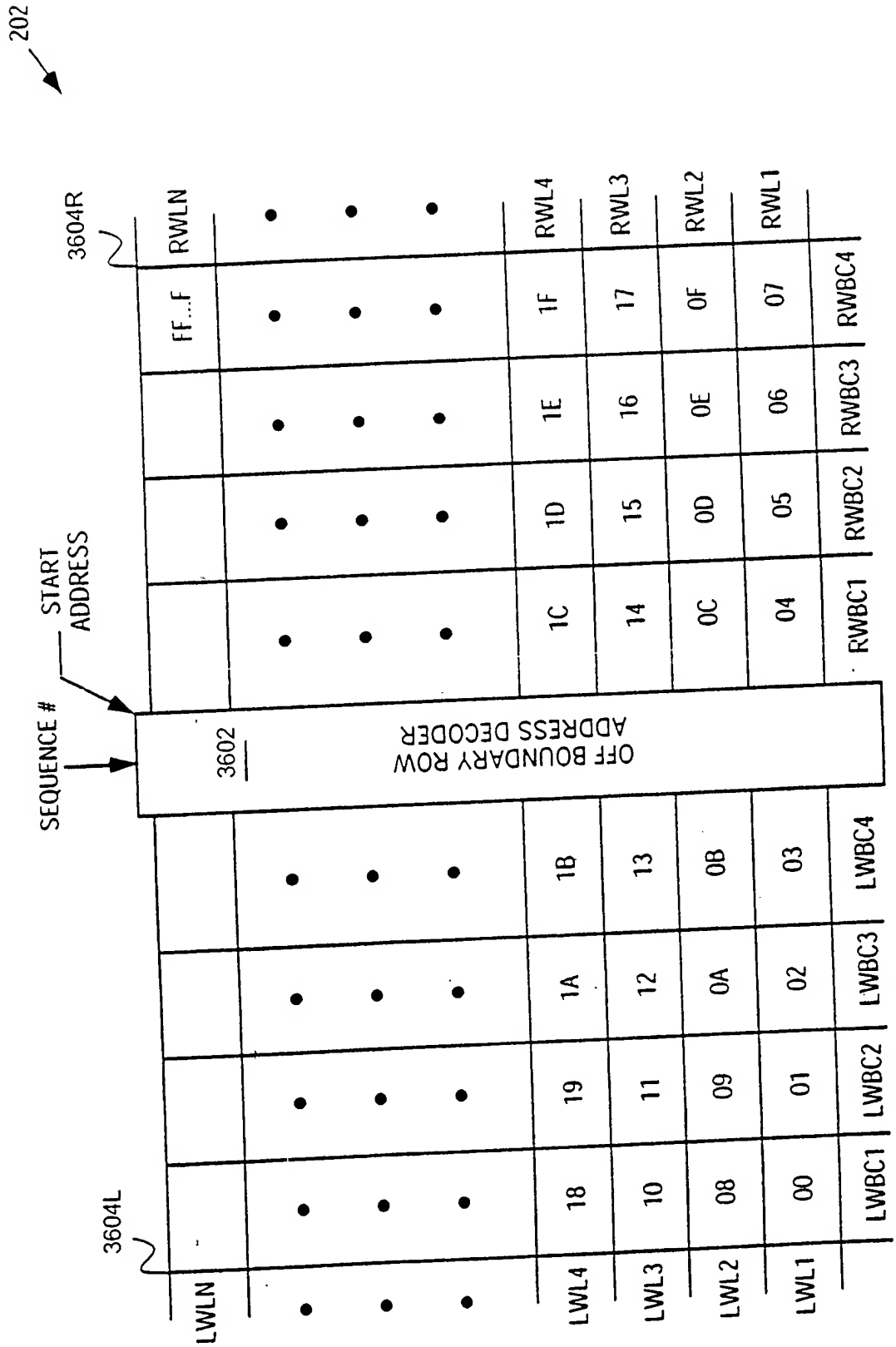

202
3604R
RWLN
FF...F
RWL4
1F
17
0F
07
RWBC4
1E
16
0E
06
RWBC3
1D
15
0D
05
RWBC2
1C
14
0C
04
RWBC1
START ADDRESS
SEQUENCE #
3602
OFF BOUNDARY ROW ADDRESS DECODER
3604L
LWLN
LWL4
1B
13
0B
03
LWBC4
1A
12
0A
02
LWBC3
19
11
09
01
LWBC2
18
10
08
00
LWBC1
LWL3
LWL2
LWL1

FIG. 36A

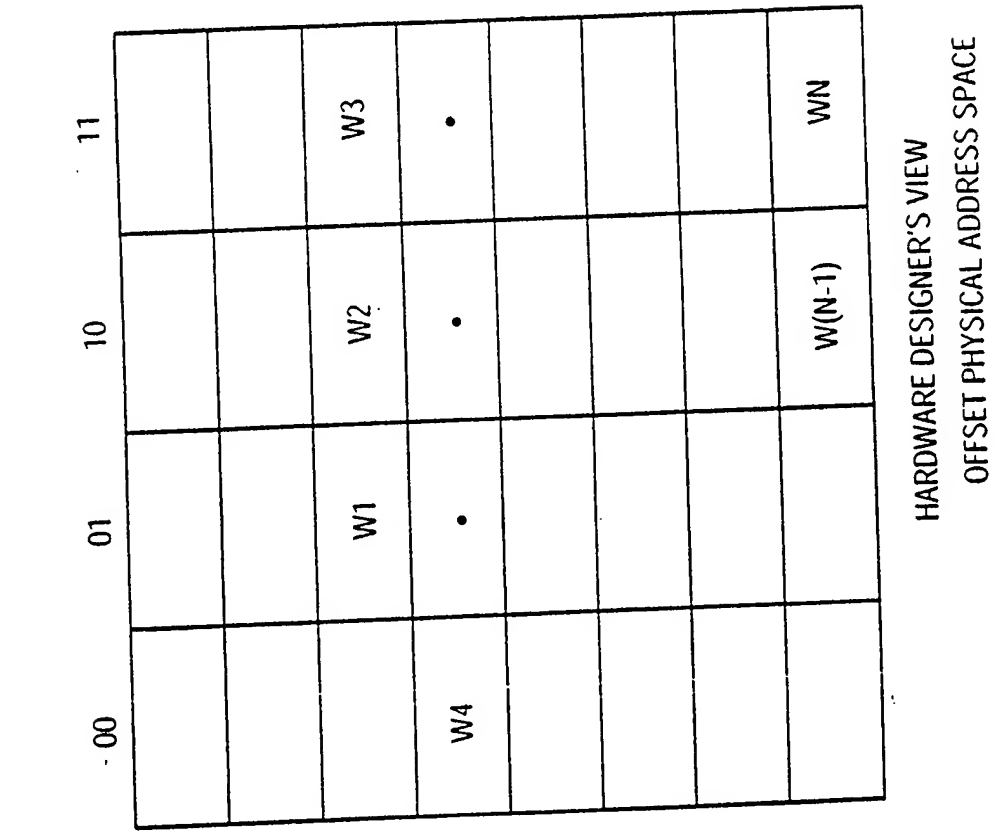


FIG. 36C

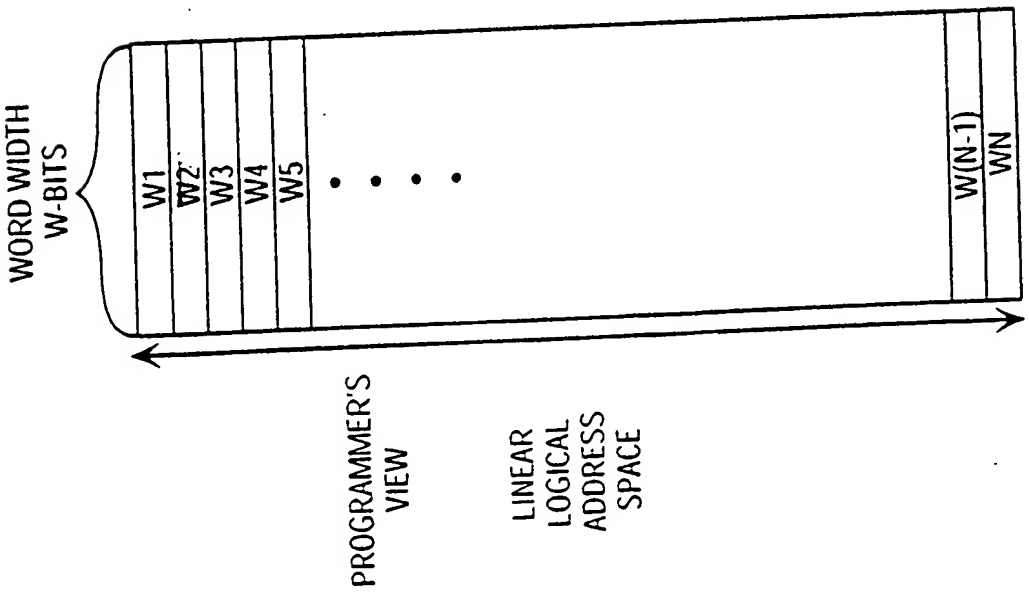


FIG. 36B

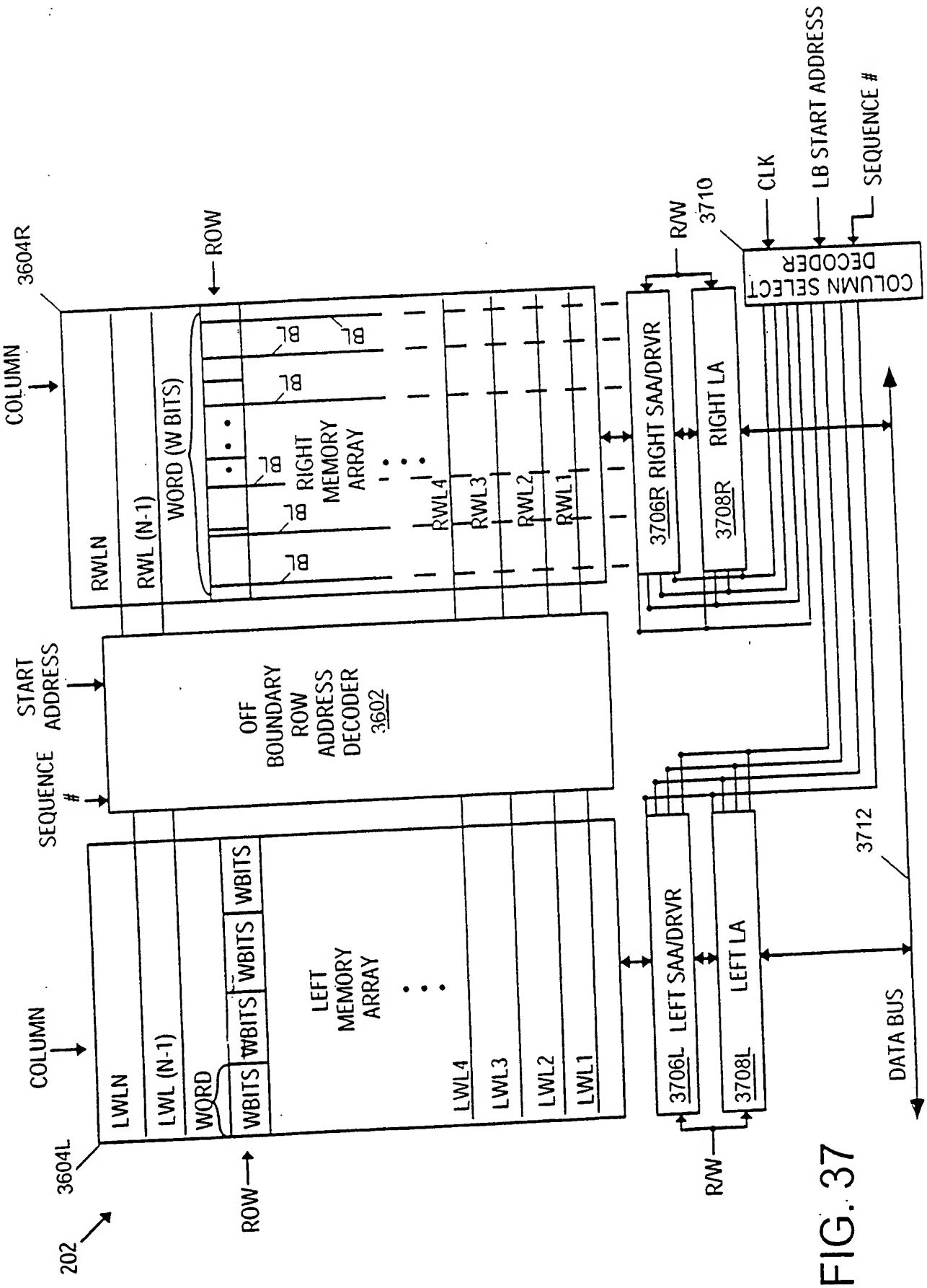

202
3604L
3604R
COLUMN
ROW
LWLN
LWL (N-1)
WORD
WBITS
LEFT MEMORY ARRAY
LWL4
LWL3
LWL2
LWL1
RWLN
RWL (N-1)
WORD (W BITS)
BL
RIGHT MEMORY ARRAY
RWL4
RWL3
RWL2
RWL1
START ADDRESS
SEQUENCE #
OFF BOUNDARY ROW ADDRESS DECODER 3602
3706L LEFT SA/DRVR
3708L LEFT LA
3706R RIGHT SA/DRVR
3708R RIGHT LA
R/W
3710
COLUMN SELECT DECODER
CLK
LB START ADDRESS
SEQUENCE #
3712
DATA BUS

FIG. 37

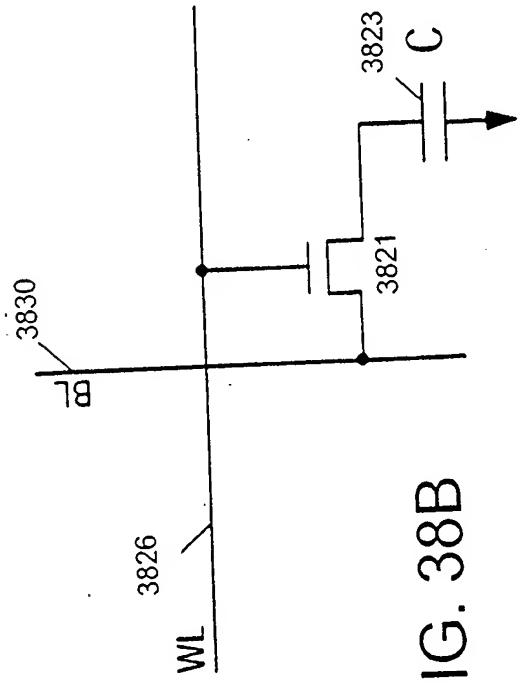


FIG. 38B

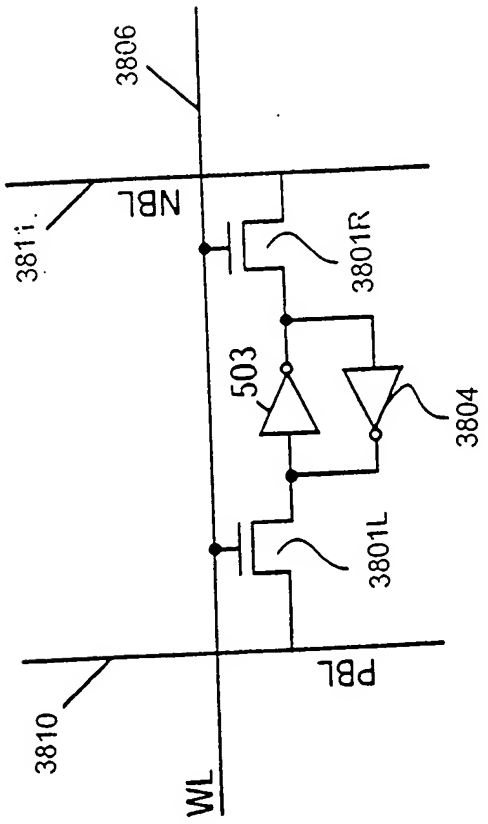


FIG. 38A

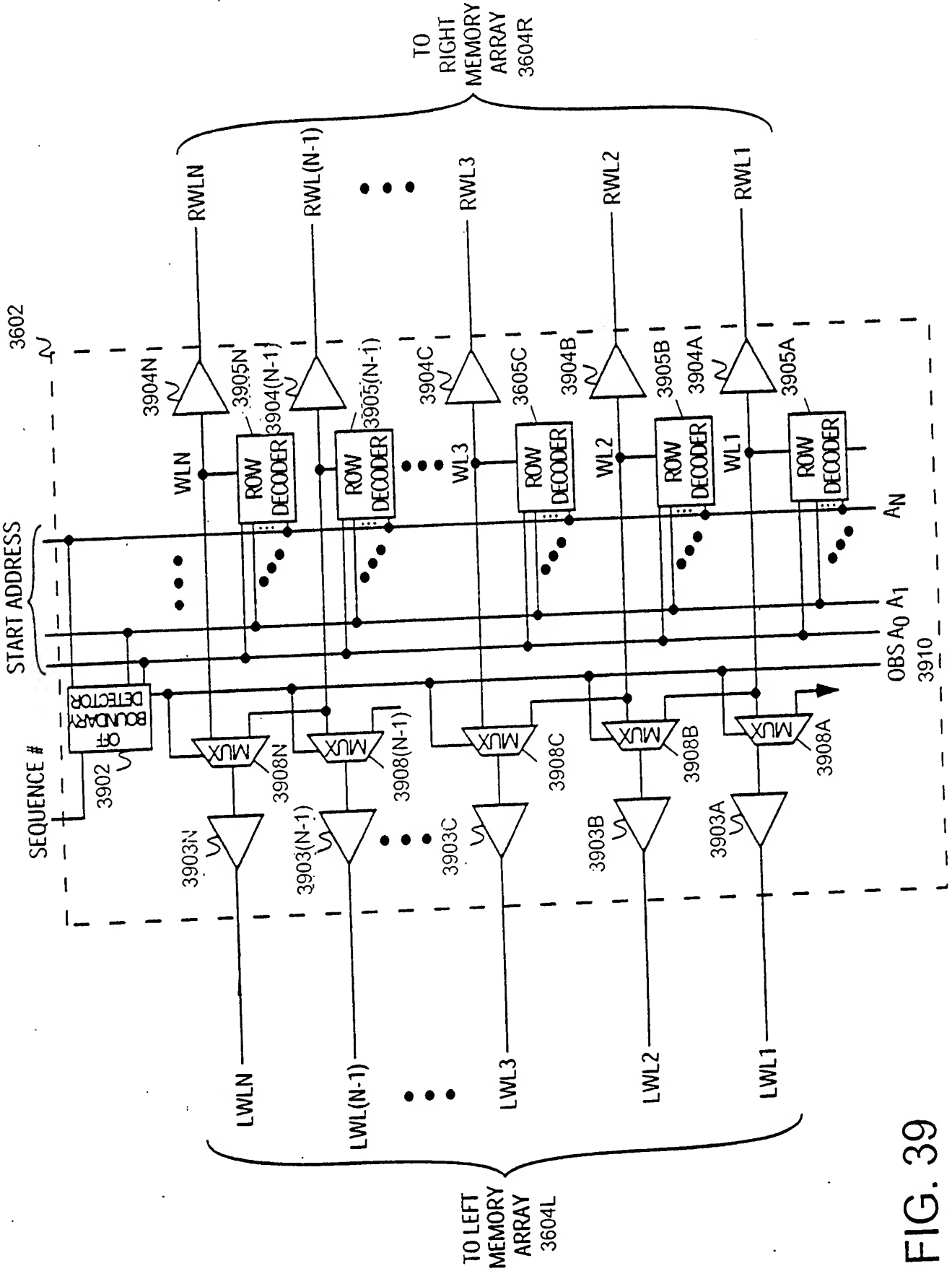

TO RIGHT MEMORY ARRAY 3604R
RWLN
RWL(N-1)
RWL3
RWL2
RWL1
3602
3904N
3905N
3904(N-1)
3905(N-1)
3904C
3605C
3904B
3905B
3904A
3905A
WLN
WL3
WL2
WL1
ROW DECODER
START ADDRESS
A_N
A_1
A_0
OBS
3910
SEQUENCE #
BOUNDARY DETECTOR OR
3902
MUX
3908N
3908(N-1)
3908C
3908B
3908A
3903N
3903(N-1)
3903C
3903B
3903A
LWLN
LWL(N-1)
LWL3
LWL2
LWL1
TO LEFT MEMORY ARRAY 3604L

FIG. 39

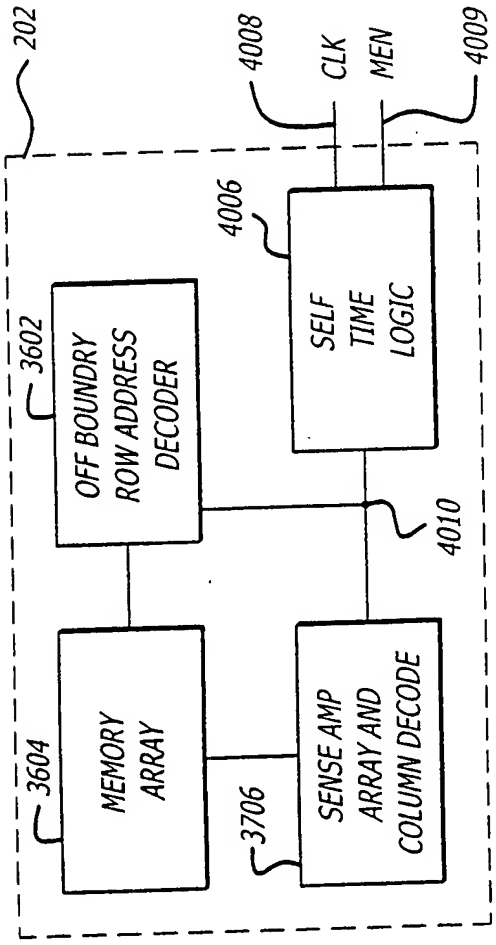


FIG. 40

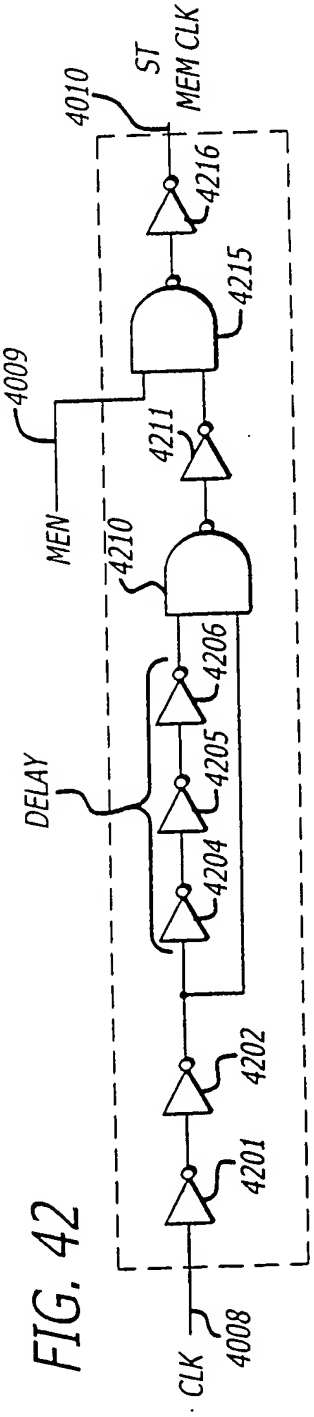


FIG. 42

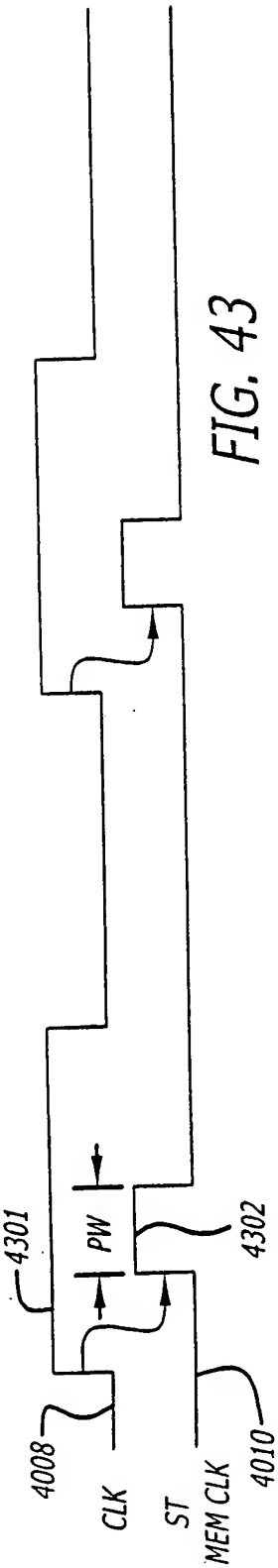
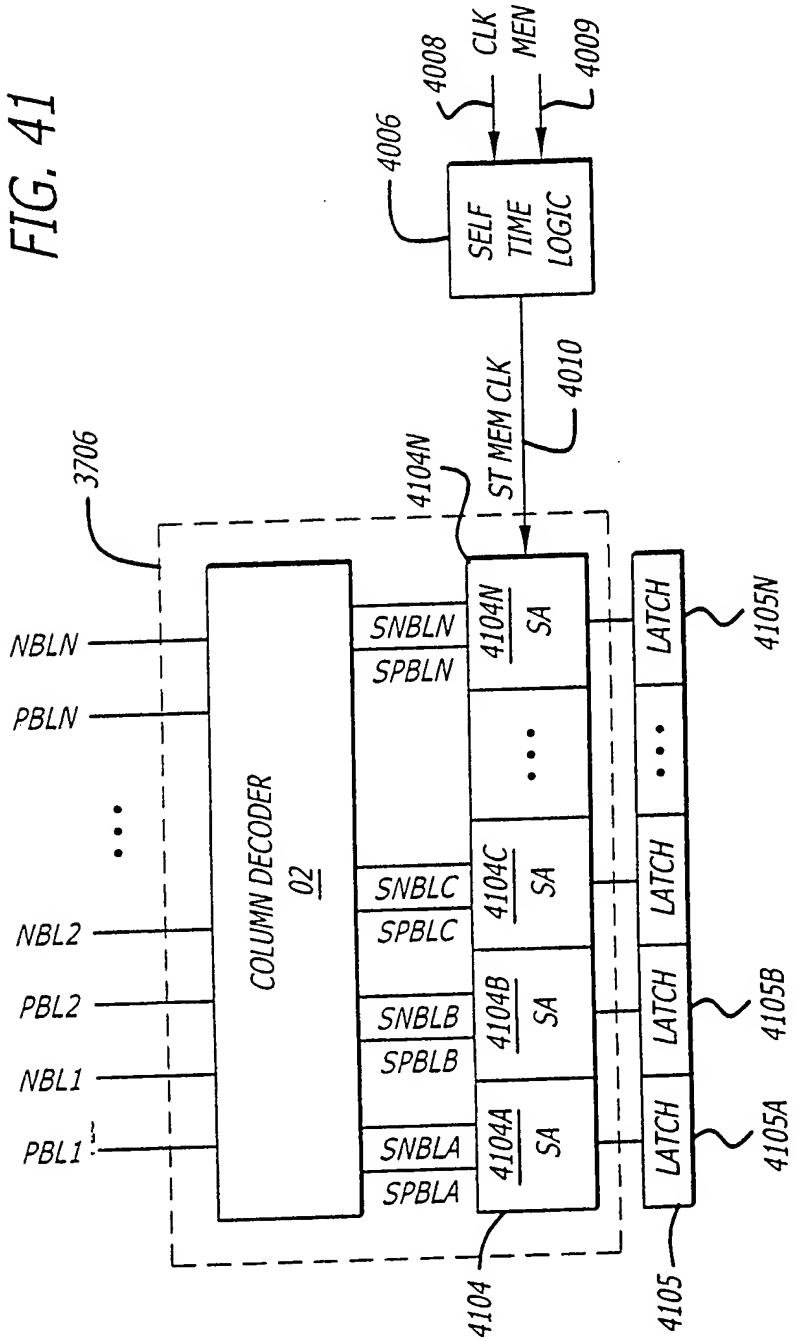


FIG. 43

FIG. 41


4008
CLK
MEN
4009
4006
SELF TIME LOGIC
4010
ST MEM CLK
3706
4104N
NBLN
PBLN
NBL2
PBL2
NBL1
PBL1
COLUMN DECODER 02
SNBLN
SPBLN
SNBLC
SPBLC
SNBLB
SPBLB
SNBLA
SPBLA
4104N SA
4104C SA
4104B SA
4104A SA
LATCH
4105N
4105B
4105A
4104
4105

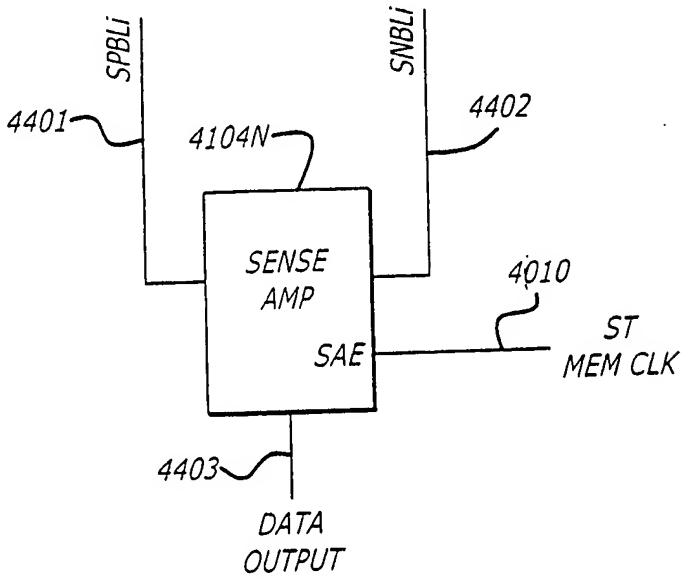


FIG. 44A

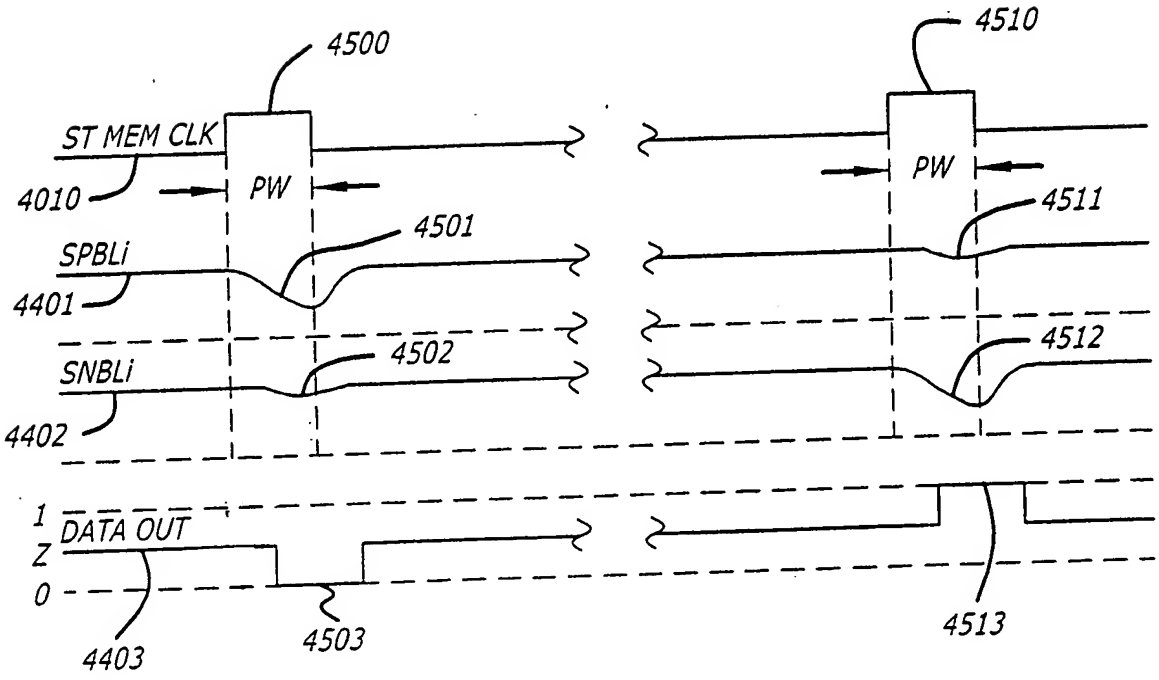


FIG. 45

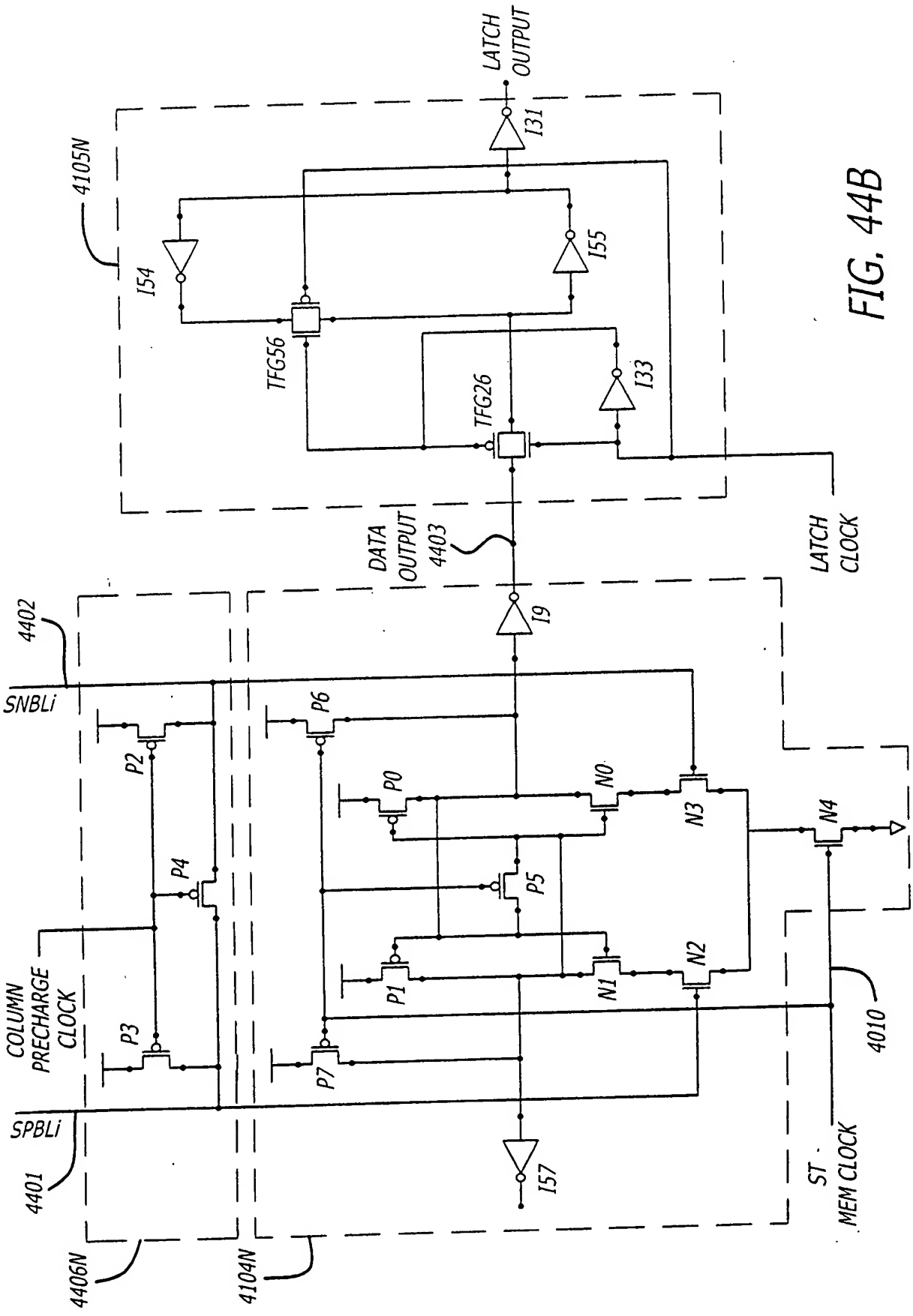

LATCH OUTPUT
I31
4105N
I54
I55
TFG56
TFG26
I33
DATA OUTPUT
4403
LATCH CLOCK
I9
4402
SNBLi
P6
P2
P0
N0
N3
N4
P4
P5
P1
N1
N2
COLUMN PRECHARGE CLOCK
P3
P7
4010
SPBLi
4401
I57
ST MEM CLOCK
4406N
4104N

FIG. 44B

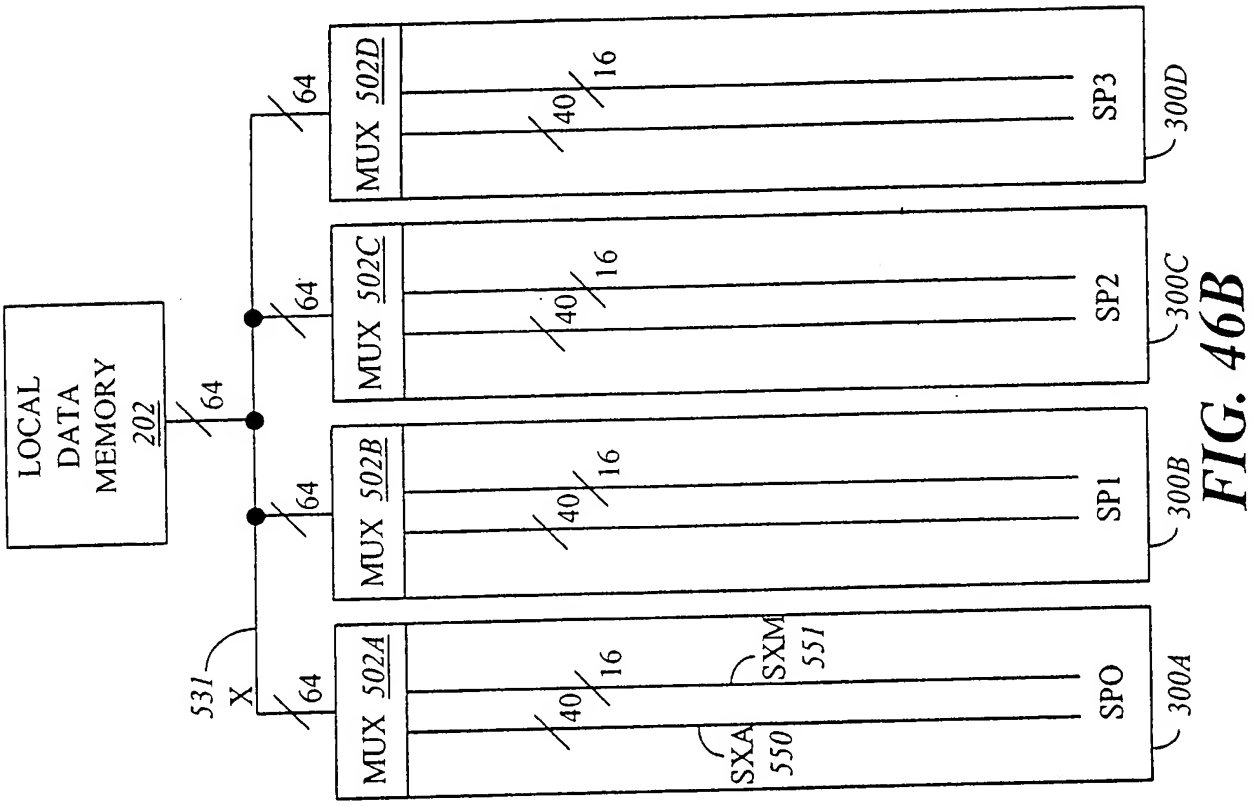


FIG. 46B

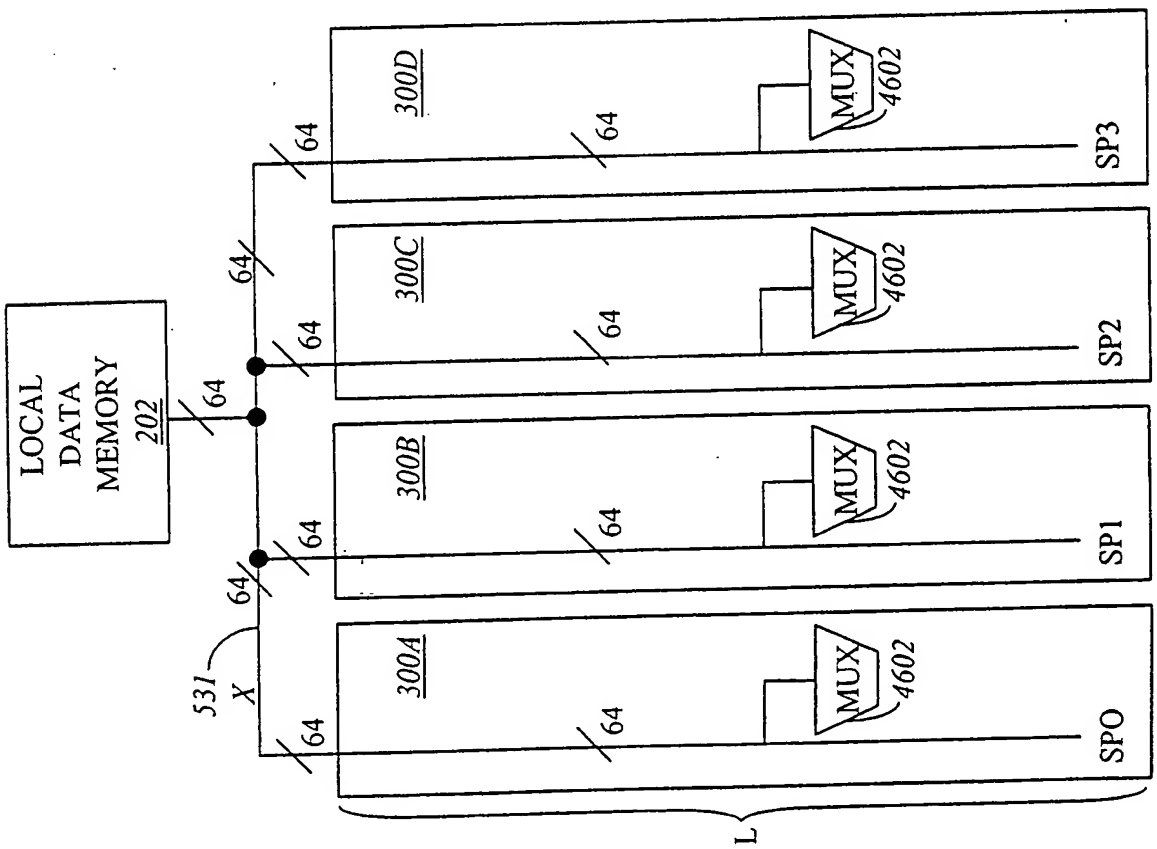


FIG. 46A